*i*INTEGRATION FRAMEWORK AND ALGORITHMS BY DESIGN:
Implicit and Explicit Families of Generalized Single Step Single Solve Algorithms by Design in Two- and Single-Field Forms

A THESIS
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Masao Shimada

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
Master of Science

Professor Kumar K. Tamma

August, 2013

Acknowledgements

I would like to thank Prof. Kumar K. Tamma and Dr. Jason Har for all their guidance and help towards completing this work. This work would not have been possible without their tremendous focus and advice.

I would also like to thank Dr. Xiangmin Zhou, Dr. Andrew Hoitink, Dr. Siti Ujila Masuri, and Vincent Wheeler for their collaboration, advice, and friendship along the way.

Dedication

To my parents and sister

Abstract

Time dependent problems are of interest here, and the designs of time integration algorithms for linear and nonlinear dynamical systems have been widely studied for the past fifty years or so and continue to be of much interest. Numerous issues need to be still resolved for transient problems so as to capture as much physics as possible and this thesis addresses these features. This thesis shows the detailed developments towards the frameworks of the generalized single step single solve (GSSSS/GS4) family of algorithms, and leading to the general frameworks comprising of families of implicit and explicit time integration schemes in the two- and single-field forms. The basic idea of designing the time integration schemes is based upon and emanates from the time weighted residual methodology. The major developments include the following considerations: (i) All the resulting developments are strictly second-order time accurate which is an important design concern, (ii) all algorithms possess a consistent time level in the discretized equations which is not well understood to-date, (iii) Linear dynamics and algorithms and designs are first addressed, (iv) the design of implicit frameworks and the corresponding predictor-corrector explicit algorithms and designs then follow; (v) how to properly extend linear dynamics algorithms to nonlinear dynamics applications is then addressed using a novel normalized time weighted residual methodology and leading to those termed as symplectic-momentum conserving and energy momentum conserving designs, and (iv) lastly, a new and novel *iIntegration* framework that is applicable to both second order systems and first order systems is finally designed for applicability to general computational engineering and science problems. The various relations to scenarios emanating from other methods of development and typical of variational algorithms and exact energy-momentum conserving algorithms to the time integration framework presented in this thesis are also carefully discussed. Both N-body systems and continuum elastodynamics

applications are illustrated and numerous numerical experiments of a wide variety of applications confirm the theoretical developments. Most of the designs of algorithms within the past 50 years or so and related to LMS methods are part of the present unified framework; and also new avenues and algorithm designs are an additional contribution including optimal designs of algorithms. Consequently, one has to simply implement the present technology which provides a wide variety of choices to the analyst in a simple setting whilst permitting to switch algorithms from one design to another based upon the problem at hand.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

The general algorithmic architectures of the single step single solve implicit/explicit schemes are proposed. It is well-known that classical mechanics is classified into three frameworks, namely, Newtonian, Lagrangian, and Hamiltonian mechanics. The Newtonian mechanics is based upon the three famous Newton's laws, and Newton's 2nd law is mathematically described as Newton's equation of motion. Lagrangian and Hamiltonian mechanics are the two common re-formulations of the Newtonian mechanics with the understanding that no new mechanics is created in these latter descriptions. Newton's equation of motion forms a set of $3N$ second-order ordinary differential equations motion, but can be also represented as a set of first-order ordinary differential equations of motion. Newton's equation of motion usually reads in the single-field form in the vectorial setting as

$$\mathbf{M}\ddot{\mathbf{x}}(t) = \mathbf{f}^{\text{appl}}(t) + \mathbf{f}^{\text{cr}}(t) \;\; \forall t \in \mathbb{I} \tag{1.1}$$

where $\ddot{\mathbf{x}}(t) = (\ddot{\mathbf{x}}^1(t), \cdots, \ddot{\mathbf{x}}^N(t)) : \mathbb{I} \to \mathbb{R}^{3N}$, $\mathbf{M} := \text{diag}(m_1, \cdots, m_N)$, $\mathbf{f}^{\text{appl}}(t) = (\mathbf{f}_1^{\text{appl}}(t), \cdots, \mathbf{f}_N^{\text{appl}}(t)) : \mathbb{I} \to \mathbb{R}^{3N}$, $\mathbf{f}^{\text{cr}}(t) = (\mathbf{f}_1^{\text{cr}}(t), \cdots, \mathbf{f}_N^{\text{cr}}(t)) : \mathbb{I} \to \mathbb{R}^{3N}$ denotes a set of the accelerations of N particles in the Cartesian coordinate system, a constant mass matrix, a set of the applied forces, and a set of the constraint forces, respectivaly. By introducing the velocity vectors, $\mathbf{v}(t) = \dot{\mathbf{x}}(t)$, Eq. (1.1) can be

represented in the two-field form as first-order ordinary differential equations, i.e.,

$$\begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{Bmatrix} \dot{\mathbf{v}}(t) \\ \dot{\mathbf{x}}(t) \end{Bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ -\mathbf{I} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \mathbf{v}(t) \\ \mathbf{x}(t) \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}^{\text{appl}}(t) + \mathbf{f}^{\text{cr}}(t) \\ \mathbf{0} \end{Bmatrix} \quad \forall t \in \mathbb{I} \tag{1.2}$$

We shall call the equations of motion represented in Eq. (1.1) and Eq. (1.2) as the single-field form and the two-field form, respectively.

Due to the presence of a constraint force, one can reduce the number of degrees of freedom (n_{dof}) and introduce the so-called generalized coordinates defined on the configuration manifold to describe the motion in the dynamical system, i.e., $\mathbf{q} \in Q \equiv \mathbb{R}^{n_{\text{dof}}} \subset \mathbb{R}^{3N}$. The motion of the system is described in terms of the generalized coordinates by the so-called Lagrange's equation of motion. Here, a scalar quantity, called Lagrangian, defined via the tangent bundle has been introduced; that is, $\mathscr{L}(\mathbf{q}, \dot{\mathbf{q}}, t) : TQ \times \mathbb{I} \to \mathbb{R}$. The Lagrangian is generally defined as the kinetic energy minus the potential energy of the system. Alternatively, Hamilton introduced a new scalar quantity, called the Hamiltonian, defined via the cotangent bundle; that is, $\mathscr{H}(\mathbf{q}, \mathbf{p}, t) : T^*Q \times \mathbb{I} \to \mathbb{R}$. It is noteworthy to mention that Lagrange's equation of motion is in the second-order representation on a n_{dof}-dimensional configuration manifold, while Hamilton's equations of motion are in the first-order representation on a $2n_{\text{dof}}$-dimensional cotangent bundle. Therefore, Lagrange's equation of motion and Hamilton's equations of motion are typically cast in the single-field form and two-field form representations, respectively.

From Newton's equation of motion, we can derive Hamilton's principle, which is sometimes referred to the principle of least action. Hamilton's principle is a variational principle, and it leads to the Lagrange's equation of motion. Although Newton's equation of motion is the basic law representing the fundamental principle of balance of linear momentum, Hamilton's principle may play as a fundamental, stand-alone principle which characterizes the physics. Hamilton's principle is valid not only for holonomic-scleronomic systems, but also for holonomic-rheonomic systems.

In contrast to these traditional practices, here we additionally introduce a

viable alternative via a new Total Energy representation of the equation of motion, and the corresponding framework in configuration manifold with particular focus upon the single-field form to enable the design of both conserving and dissipative algorithms with improved physical insight and computationally attractive features. Suppose we confine our interests to a system free from constraints, and assume the system is holonomic-scleronomic. In this case, although the principle of balance of mechanical energy can be derived from Newton's equation of motion, this differential principle (in contrast to employing variational calculus) may be also used as a stand-alone principle just like the Hamilton's principle. In general, we define the total mechanical energy as the summation of the kinetic energy and the potential energy, and it is defined via the tangent bundle, similar to the Lagrangian: $\mathscr{E}(\mathbf{q}, \dot{\mathbf{q}}, t) : TQ \times \mathbb{I} \to \mathbb{R}$. Note the autonomous Hamiltonian is equivalent with the total energy in holonomic-scleronomic systems; $\mathscr{E}(\mathbf{q}, \dot{\mathbf{q}}) \equiv \mathscr{H}(\mathbf{q}, \mathbf{p})$. Similar to the Lagrangian and Hamiltonian, the total mechanical energy also has symmetries in time and translational and rotational motions.

The contributions and a brief outline of this thesis are as follows:

Chapter 2: The derivation of the equations of motion in the Newtonian, Lagrangian, and Hamiltonian mechanics via the variational and differential approaches are briefly summarized. The vector and scalar formalisms are presented.

Chapter 3: The derivations of the strong and weak forms of Cauchy's first equation of motion in the sense of both the vector and scalar formalisms are discussed. The derivation of Cauchy's first equation of motion in the sense of the vector formalism, namely, from the principle of balance of total linear momentum for a continuum body and the derivation of its semi-discrete form following the Galerkin finite element method are shown. In the scalar-formalisms, unlike the vector formalism, we mainly deal with the scalar-valued functions; that is, the autonomous Lagrangian function and the autonomous total energy function of an isothermal

continuum body, which is sometimes called the autonomous mechanical energy in the literature; and the autonomous Hamiltonian function, which is equivalent with the autonomous mechanical energy function. The key point that both Hamilton's principle and the principle of balance of mechanical energy are consequences of Cauchy's first equation of motion is emphasized. Once the Cauchy's first equation of motion in the Lagrangian and Hamiltonian mechanics is derived, the spatially-discrete forms are readily obtained following the Galerkin finite element method.

Chapter 4: Unified time integration frameworks including single-step single-solve implicit algorithms obtained by the time discretizations of the linear semi-discrete equations of motion are first discussed. This framework is termed as the family of the ***generalized single step single solve (GSSSS) algorithms*** and has been designed such that they possess: (1) the necessary second-order time accuracy, (2) unconditional stability, (3) zero-order overshoot behavior(s) in both/either configuration and/or velocity, and (4) controllable numerical dissipation features. It is well-known that the GSSSS algorithmic framework includes various existing direct time integration algorithms such as numerically-non-dissipative Newmark method and the implicit midpoint rule, and controllable numerically-dissipative methods such as HHT-, WBZ, and SSpj methods, the three parameter optimal scheme [4,5] (or equivalently the Generalized-α method) [3] etc. More noteworthy, the GSSSS (or GS4) family of algorithms and designs additionally contain more newer algorithms and optimal algorithms as well in contrast to existing state-of-the-art! The theory underlying GS4 algorithms in linear dynamical systems plays a basic role when we discretize the (semi-discrete) equations of motion for extensions to nonlinear dynamical systems.

Chapter 5: The design of predictor-corrector explicit family of the GSSSS algorithms are first discussed for the linear dynamical system. All the schemes in this framework maintain the necessary second-order time accuracy. Furthermore, we also discuss the general explicit GSSSS algorithm designs developed for the

second-order system which additionally contain the algorithm designs over and beyond the predictor-corrector GSSSS family of algorithms. The underlying reason is that we relax the conditions of unconditional stability and invoke conditional stability for all the explicit GSSSS general structure of algorithms and designs.

Chapter 6: The extensions of the implicit family of the GSSSS algorithms in linear dynamical systems, discussed in Chapter 4, via the so-called normalized time weighted residual methodology is carefully discussed for applications to nonlinear dynamical systems. In particular, we draw attention to the classical time weighted residual methodology (Option I) and the new normalized time weighted residual methodology (Option II). The extension to nonlinear dynamical systems by the normalized time weighted residual methodology in the sense of Option II leads to the symplectic-momentum conserving/dissipative framework of the GSSSS family of algorithms. We can adjust the numerical dissipation of the algorithms simply by controlling the algorithmic parameters (***controllable numerical dissipation features***).

Chapter 7: Starting from and after describing the principle of balance of mechanical energy in the single-field form which employs differential calculus instead of variational calculus, we start the discretization process via two distinct approaches. The first approach is based upon the mean-value theorem to design exact energy and energy-momentum conserving algorithms. And, the second approach is based upon a *hybrid* normalized time weighted residual methodology. We discuss the relation between the implicit family of the GSSSS algorithms from the first approach and the energy-momentum conserving/dissipative schemes (Option III) via the latter approach. The energy-momentum conserving/dissipative framework of the GSSSS family of algorithms also possesses the controllable numerical dissipation features.

Chapter 8: The extensions of the explicit family of the GSSSS algorithms, discussed in Chapters 5 and 6, via the so-called normalized time weighted residual methodology is carefully discussed for applications to nonlinear dynamical systems.

Chapter 9: *i*INTEGRATION Framework: All the algorithms discussed in the previous chapters are combined together in a unified setting under the umbrella of the *i*INTEGRATION framework. Consequent adaptation from algorithms and designs in the second-order systems to algorithms and designs in the the first-order systems is particularly addressed.

Chapter 2

Equations of Motion in Continuous-Time Systems: System of N Particles

2.1 Vector Formalism: Equations of Motion for a System of N Particles in Newtonian Mechanics, and Consequent Transformation to Generalized Coordinates Leading to Lagrangian, and Hamiltonian Mechanics

2.1.1 Newton's Equation of Motion

For convenience, we select a Cartesian coordinate frame for $\mathbb{E}^3$, i.e., an origin $\mathbf{o} \in \mathbb{E}^3$ with a right-handed orthonormal basis $\{\mathbf{e}_i\}_{i=1}^3 = \{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$ for a vector space $\mathcal{V}$, as an inertial frame of reference in three dimensional Euclidean space to describe the three-dimensional motion of a particle or an N-body dynamical system in Newtonian mechanics. By a right-handed orthonormal basis $\{\mathbf{e}_i\}_{i=1}^3$ for

$\mathcal{V}$, we mean

$$\mathbf{e}_1 \times \mathbf{e}_2 = \mathbf{e}_3, \quad \mathbf{e}_2 \times \mathbf{e}_3 = \mathbf{e}_1, \quad \mathbf{e}_3 \times \mathbf{e}_1 = \mathbf{e}_2 \tag{2.1}$$

and

$$(\mathbf{e}_1 \times \mathbf{e}_2) \cdot \mathbf{e}_3 = 1 \tag{2.2}$$

Note that the inertial frame of reference in Newtonian mechanics can also be curvilinear coordinate systems such as the cylindrical or spherical polar coordinate systems. Assume a moving particle, a material point $\mathbf{x} \in \mathbb{E}^3$ is of mass $m \in \mathbb{R}^+$. Note the position vector is $\mathbf{x} - \mathbf{o} \in \mathcal{V}$. The coordinates of $\mathbf{x}$ are given by

$$x^i = (\mathbf{x} - \mathbf{o}) \cdot \mathbf{e}_i = \mathbf{x} \cdot \mathbf{e}_i \in \mathbb{R} \text{ for } i = 1, 2, 3 \tag{2.3}$$

Hence, x^i $(i = 1, 2, 3)$ are uniquely determined as

$$\mathbf{x} - \mathbf{o} = x^1\mathbf{e}_1 + x^2\mathbf{e}_2 + x^3\mathbf{e}_3 \tag{2.4}$$

In the matrix representation, $[\mathbf{x}-\mathbf{o}] = (x^1-0, x^2-0, x^3-0)^T = (x^1, x^2, x^3)^T \in \mathbb{R}^3$ where $\mathbb{R}^3$ denotes all ordered triples of real numbers. For simplicity, we write the position vector as $\mathbf{x}$ instead of $\mathbf{x} - \mathbf{o}$, and $\mathbf{x} = (x^1, x^2, x^3)^T \in \mathbb{R}^3$. The moving particle is completely described by the trajectory

$$\mathbf{x} = \mathbf{x}(t) \tag{2.5}$$

where time $t \in \mathbb{I} = [t_0, t_1]$ $(t_0 \geq 0)$ is the independent variable. According to Newton's second law of motion, the particle obeys the following second-order differential equation with given initial position $\mathbf{x}(t_0)$ and initial velocity $\dot{\mathbf{x}}(t_0) := (d/dt)\mathbf{x}(t_0)$:

$$m\frac{d^2}{dt^2}\mathbf{x}(t) = \mathbf{f}^{\text{total}}(\mathbf{x}(t), t) \tag{2.6}$$

with the constant mass of the particle, $\dot{m} = 0$, where $\mathbf{f}^{\text{total}}$ denotes a total resultant force acting on the point. Newton's second law can be generalized as the principle of balance of momentum by

$$\frac{d}{dt}\mathbf{p} = \mathbf{f}^{\text{total}}(\mathbf{x}, \dot{\mathbf{x}}, t) \tag{2.7}$$

where $\mathbf{p}$ denotes the momentum. If the momentum is given by $\mathbf{p} = m\dot{\mathbf{x}}$ and the force depends upon the instantaneous position and time as $\mathbf{f}^{\text{total}}(\mathbf{x}(t), t)$, Eq. (2.7) leads to Eq. (2.6). An example of the force dependent upon the velocity may be the Lorentz on a point charge due to electromagnetic fields:

$$\mathbf{f}^{\text{total}} = e\left(\mathbf{E} + \frac{\mathbf{v}}{c} \times \mathbf{B}\right) \tag{2.8}$$

with $\dot{\mathbf{x}} = \mathbf{v}$ where e, $\mathbf{E}$, and $\mathbf{B}$ denote the electric charge of the particle, the electric field, and the magnetic field, respectively. The momentum $\mathbf{p}$ does not need to be linear in the velocity. For example, in relativistic mechanics, the momentum may be replaced with the relativistic momentum, $\mathbf{p} = m(\dot{\mathbf{x}})\dot{\mathbf{x}}$, with the relativistic mass given by

$$m(\dot{\mathbf{x}}) = m_0 / \sqrt{1 - \frac{\dot{\mathbf{x}} \cdot \dot{\mathbf{x}}}{c^2}} \quad (m \geq m_0) \tag{2.9}$$

where m_0 and c denotes the mass in its rest frame (the rest mass) and the speed of light, respectively.

Next, consider a system of N particles moving in the Cartesian coordinate frame for $\mathbb{E}^3$. The principle of balance of momentum for each particle with $\mathbf{p}_a = m_a \dot{\mathbf{x}}^a$ and $\mathbf{f}_a^{\text{total}}(\mathbf{x}, t)$,

$$\dot{m}_a = 0 \ \text{ and } \ \frac{d}{dt}\mathbf{p}_a = \mathbf{f}_a^{\text{total}}(\mathbf{x}, t) \ \ \forall a = 1, 2, \ldots, N \tag{2.10}$$

leads to Newton's equation of motion governs the motion of the a^{th} particle,

$$\dot{m}_a = 0 \ \text{ and } \ m_a \frac{d^2}{dt^2}\mathbf{x}_a(t) = \mathbf{f}_a^{\text{total}}(\mathbf{x}(t), t) \ \ \forall a = 1, 2, \ldots, N \tag{2.11}$$

where $m_a \in \mathbb{R}$, $\mathbf{x}_a \in \mathbb{R}^3$, and $\mathbf{f}_a^{\text{total}} \in \mathbb{R}^3$ denotes a constant mass of the a^{th} particle, the position vector at $t \in \mathbb{I}$, and the total resultant force acting on the a^{th} particle, respectively. Note that the position and the total resultant force vectors on the a^{th} particle in $\mathbb{E}^3$ can be strictly written as $\mathbf{x}_a - \mathbf{o} \in \mathcal{V}$ and $\mathbf{f}_a^{\text{total}} - \mathbf{o} \in \mathcal{V}$, respectively. A trajectory of the system in the three-dimensional motion is a curve of a vector-valued function of the $3N$-number of multi-variables as a set, i.e.,

$$\begin{aligned} \mathbf{x}(t) &= \left(\mathbf{x}^1(t), \cdots, \mathbf{x}^N(t)\right)^T \\ &= \left(x^1(t), x^2(t), x^3(t), \cdots, x^{3N}(t)\right)^T : \mathbb{I} \to \mathbb{R}^{3N}, \ \forall t \in \mathbb{I} \end{aligned} \tag{2.12}$$

where the range of the vector-valued function of $3N$ variables is contained in Euclidean $3N$-space which is defined as $\mathbb{R}^{3N} := \mathbb{R}^3 \times \mathbb{R}^3 \times \cdots \times \mathbb{R}^3$. Similarly, we define $\mathbf{f}^{\text{total}} = (f_1^{\text{total}}, f_2^{\text{total}}, \cdots, f_{3N}^{\text{total}}) \in \mathbb{R}^{3N}$. Hence, Eq. (2.11) can be written as

$$\dot{\mathbf{M}} = \mathbf{0} \text{ and } \mathbf{M}\frac{d^2}{dt^2}\mathbf{x}(t) = \mathbf{f}^{\text{total}}(\mathbf{x}, t) \tag{2.13}$$

with the diagonal mass matrix $[\mathbf{M}] = \text{diag}(m, m, m, \cdots, m) \in \mathbb{R}^{3N} \times \mathbb{R}^{3N}$. Suppose the total resultant force $\mathbf{f}_a^{\text{total}}$ is given by the form of $\mathbf{f}_a^{\text{total}}(\mathbf{x}(t), t)$ and initial conditions are also given; then, the N number of Equations (2.11) are simultaneous second-order differential equations of the N-body dynamical system. The total resultant force $\mathbf{f}_a^{\text{total}}$ can be divided into two parts: $\mathbf{f}_a^{\text{total}} = \mathbf{f}_a^{\text{ext}} + \mathbf{f}_a^{\text{int}}$ where $\mathbf{f}_a^{\text{ext}}(t) : \mathbb{I} \subset \mathbb{R}^+ \to \mathbb{R}^3$ denotes the external force acting upon the a^{th} particle and $\mathbf{f}_a^{\text{int}}(t) : \mathbb{R} \to \mathbb{R}^3$ denotes the internal force as the summation of the *interactions* acting on the a^{th} particle owing to the b^{th}, i.e., $\mathbf{f}_a^{\text{int}} = \sum_{b=1}^{N} \mathbf{f}_{ab}^{\text{int}}$. It is important to note Newton's third law results in the total internal force, i.e., the internal force in the system; that is,

$$\sum_{a=1}^{N} \mathbf{f}_a^{\text{int}}(t) = \sum_{a=1}^{N}\sum_{b\neq a}^{N} \mathbf{f}_{ab}^{\text{int}} = \mathbf{0} \tag{2.14}$$

because $\mathbf{f}_{ab}^{\text{int}} = -\mathbf{f}_{ba}^{\text{int}}$. Hence, the equations of the three-dimensional motion of the N-body system yields

$$\frac{d}{dt}\left(\sum_{a=1}^{N} m_a \dot{\mathbf{x}}^a\right) = \sum_{a=1}^{N} m_a \ddot{\mathbf{x}}^a = \sum_{a=1}^{N} \mathbf{f}_a^{\text{ext}} =: \mathbf{F}^{\text{ext}} \tag{2.15}$$

where $\mathbf{F}^{\text{ext}}(t) : \mathbb{I} \to \mathbb{R}^3$ is the total external force of the system as a vector-valued function of one real variable t. The total external force, which is often called the resultant force, may be dependent upon the position, the velocities, and time, but it can not be a function of acceleration. It is naturally known that $\mathbf{x} = (x^1, x^2, \ldots, x^{3N})$ are related each other via the constraint conditions. The holonomic constraint condition, for example, is described by functions

$$f_\tau(\mathbf{x}, t) = 0 \tag{2.16}$$

for $\tau = 1, 2, \ldots, s$. The constraint condition is called the "holonomic-rheonomic" constraint when f_τ depends on time t explicitly, whereas the constraint condition is called the "holonomic-scleronomic" constraint when f_τ does NOT depend on time t explicitly:

$$f_\tau\left(\mathbf{x}\left(t\right), t\right) = 0; \text{ Holonomic-Rheonomic Constraint} \tag{2.17}$$

$$f_\tau\left(\mathbf{x}\left(t\right)\right) = 0; \text{ Holonomic-Scleronomic Constraint} \tag{2.18}$$

for $\tau = 1, 2, \ldots, s$. It is important to note that the exact differentials of Eq. (2.17) and (2.18) become

$$df_\tau = \sum_{a=1}^{N} \left(\frac{\partial f_\tau}{\partial \mathbf{x}^a} \cdot d\mathbf{x}^a \right) + \frac{\partial f_\tau}{\partial t} dt = 0 \tag{2.19}$$

$$df_\tau = \sum_{a=1}^{N} \left(\frac{\partial f_\tau}{\partial \mathbf{x}^a} \cdot d\mathbf{x}^a \right) = 0 \tag{2.20}$$

respectively. Notice that the vector normal to the constraint surface is NOT orthogonal to the differential of the position vector in holonomic-rheonomic constraints, whereas it is orthogonal to the differential of the position vector in holonomic-scleronomic constraints. Suppose that the N-body dynamical system is now subject to some constraints which limit the motion of the system in the Cartesian coordinate system. Then, the external force exerted on the a^{th} particle is divided into the applied force and the constraint force: $\mathbf{f}_a^{\text{ext}} = \mathbf{f}_a^{\text{appl}} + \mathbf{C}_a$ where $\mathbf{C}_a$ is the constraint force acting upon the a^{th} particle and $\mathbf{f}_a^{\text{appl}}\left(t\right) : \mathbb{I} \to \mathbb{R}^3$ is the applied force vector acting upon the a^{th} particle as a vector-valued function of the variable t. The applied force $\mathbf{f}_a^{\text{appl}}$ can be divided in two parts: $\mathbf{f}_a^{\text{appl}} = \mathbf{f}_a^{\text{c}} + \mathbf{f}_a^{\text{nc}}$ where $\mathbf{f}_a^{\text{c}}$ and $\mathbf{f}_a^{\text{nc}}$ denote the conservative force and the non-conservative force acting upon the a^{th} particle. Next, consider the potential energy defined for the motion of a system of N particles in a potential field in Euclidean $3N$-space as follows:

$$\mathscr{U}\left(\mathbf{x}\right) := \mathscr{U}^{\text{int}} + \mathscr{U}^{\text{ext}} \tag{2.21}$$

where the total internal potential energy of interactions between a^{th} and b^{th} particles and the total external potential energy of a^{th} particle are given by

$$\mathscr{U}^{\text{int}} := \sum_{a=1}^{N} \sum_{b>a}^{N} \mathscr{U}_{ab}^{\text{int}}(\mathbf{x}^a, \mathbf{x}^b) \;\text{ and }\; \mathscr{U}^{\text{ext}} := \sum_{a=1}^{N} \mathscr{U}_{a}^{\text{ext}}(\mathbf{x}^a) \tag{2.22}$$

respectively. The conservative force $\mathbf{f}_a^{\text{c}}$ acting upon the a^{th} particle is defined by the negative gradient of the external potential energy, i.e.,

$$\mathbf{f}_a^{\text{c}} = -\frac{d\mathscr{U}_a^{\text{ext}}(\mathbf{x}^a)}{d\mathbf{x}^a} = \left(f_{xa}^c, f_{ya}^c, f_{za}^c\right) \tag{2.23}$$

Note that the non-conservative force acting upon the a^{th} particle cannot be expressed in terms of any potential energy. Similarly, from the internal potential energy, we define the internal force due to the *interactions* between a^{th} and b^{th} particles by

$$\mathbf{f}_a^{\text{int}} = -\sum_{b\neq a}^{N} \frac{\partial \mathscr{U}_{ab}^{\text{int}}(\mathbf{x}^a, \mathbf{x}^b)}{\partial \mathbf{x}^a} \tag{2.24}$$

Recall the total internal force vanishes; see Eq. (2.14). If we can express the internal potential energy with respect to the inter-particle distance, i.e., $r_{ab} := \| \mathbf{x}^b - \mathbf{x}^a \| := \sqrt{(\mathbf{x}^b - \mathbf{x}^a) \cdot (\mathbf{x}^b - \mathbf{x}^a)}$, the total internal potential energy may be written as

$$\mathscr{U}^{\text{int}} = \sum_{a=1}^{N} \sum_{b>a}^{N} \mathscr{U}_{ab}^{\text{int}}(r_{ab}) \tag{2.25}$$

and Eq. (2.24) becomes

$$\mathbf{f}_a^{\text{int}} = -\sum_{b\neq a}^{N} \frac{d\mathscr{U}_{ab}^{\text{int}}(r_{ab})}{dr_{ab}} \frac{\mathbf{x}^{ab}}{r_{ab}} \tag{2.26}$$

Newton's second law of motion of the N-body dynamical system can be written

as

$$\begin{aligned}\sum_{a=1}^{N} m_a \frac{d^2\mathbf{x}^a(t)}{dt^2} &= \sum_{a=1}^{N} \mathbf{f}_a^{\text{total}} = \sum_{a=1}^{N} \mathbf{f}_a^{\text{ext}} + \underbrace{\sum_{a=1}^{N} \mathbf{f}_a^{\text{int}}}_{\mathbf{0}} \\ &= \sum_{a=1}^{N} \mathbf{f}_a^{\text{appl}} + \sum_{a=1}^{N} \mathbf{C}_a \\ &= \sum_{a=1}^{N} \mathbf{f}_a^{\text{c}} + \sum_{a=1}^{N} \mathbf{f}_a^{\text{nc}} + \sum_{a=1}^{N} \mathbf{C}_a \end{aligned} \tag{2.27}$$

It should be now observed that Equation (2.27) represents the $3N$-number of simultaneous differential equations of motion, which are redundant because of the presence of the constraints. In other words, the total number of degrees of freedom becomes smaller than the number of dependent variables, i.e., the Cartesian coordinate variables. In general, the constraints are given by algebraic equations with coordinate variables in the inertial frame of reference. As a consequence, the Newtonian dynamical system is subject to the constraint equations. Suppose that the k-number of constraints ($s = k < N$) are imposed in the system and these constraint equations are associated with Cartesian coordinate variables. Then, the number of degrees of freedom n_{dof} for the N-body dynamical system with k-constraints becomes $3N - k$, i.e., $n_{\text{dof}} = 3N - k$.

Theorem 1 (Principle of Balance of Total Linear Momentum)
Newton's equation of motion indicates the balance of the total linear momentum $\mathbf{L}$ *of a system of* N *particles:*

$$\frac{d\mathbf{L}}{dt} = \sum_{a=1}^{N} \mathbf{f}_a^{\text{ext}} \tag{2.28}$$

where $\mathbf{L} := \sum_{a=1}^{N} m_a \dot{\mathbf{x}}^a$.

Proof. Consider Newton's equation of motion for a system of particles, i.e.,

$$\sum_{a=1}^{N} m_a \frac{d^2\mathbf{x}^a(t)}{dt^2} = \sum_{a=1}^{N} \mathbf{f}_a^{\text{total}} = \sum_{a=1}^{N} \mathbf{f}_a^{\text{ext}} + \sum_{a=1}^{N} \sum_{b \neq a}^{N} \mathbf{f}_{ab}^{\text{int}} = \sum_{a=1}^{N} \mathbf{f}_a^{\text{ext}} \tag{2.29}$$

The left-hand side of Eq. (2.29) can be written in terms of the total linear momentum:

$$\text{LHS} = \sum_{a=1}^{N} \frac{d^2 m_a \mathbf{x}^a}{dt^2} = \sum_{a=1}^{N} \frac{d\left(m_a \dot{\mathbf{x}}^a\right)}{dt} = \frac{d}{dt}\left(\sum_{a=1}^{N} m_a \dot{\mathbf{x}}\right) = \frac{d\mathbf{L}}{dt} \tag{2.30}$$

Therefore, Eq. (2.29) yields

$$\frac{d\mathbf{L}}{dt} = \sum_{a=1}^{N} \mathbf{f}_a^{\text{ext}} \tag{2.31}$$

Notice that the left-hand side and right-hand side of Eq. (2.31) indicate the rate of the total linear momentum and the resultant force of the dynamical system of N particles, respectively. ■

From Principle 1, the conservation of linear momentum of a system of N particles is readily obtained:

Corollary 1 (Principle of Conservation of Total Linear Momentum) *In the absence of the external force, i.e.,* $\mathbf{f}_a^{\text{ext}} = \mathbf{0}$*, the total linear momentum* $\mathbf{L}$ *of a system of* N *particles is conserved.*

$$\frac{d\mathbf{L}}{dt} = \frac{d}{dt}\left(\sum_{a=1}^{N} m_a \dot{\mathbf{x}}\right) = \mathbf{0} \tag{2.32}$$

2.1.2 Lagrange's Equation of Motion and Total Energy Representation of Equation of Motion

Consider holonomic dynamical systems where the constraint condition is given by Eq. (2.16). According to the inverse function theorem, if the following is true at $\mathbf{x}$ which satisfy the constraint condition, Eq. (2.16),

$$\text{Rank}\left[\frac{\partial\left(f_1, f_2, \ldots, f_s\right)}{\partial\left(x^1, x^2, \ldots, x^{3N}\right)}\right] = k \leq s, \tag{2.33}$$

then $\mathbf{x} = \{x^1, x^2, \ldots, x^{3N}\}$ are not independent each other, and we can define n_{dof} independent variables called the generalized coordinates $\mathbf{q} = (q^1, q^2, \ldots, q^{\text{n}_{\text{dof}}})$. Using the generalized coordinates, we have

$$\mathbf{x}^a \;=\; \mathbf{x}^a\left(\mathbf{q}, t\right) = \mathbf{x}^a\left(q^1, q^2, \ldots, q^{\text{n}_{\text{dof}}}, t\right) : Q \times \mathbb{I} \to \mathbb{R}^3 \tag{2.34}$$

where the configuration manifold Q is defined as

$$Q := \left\{ (\mathbf{q}) \mid \mathbf{q} = (q^1, q^2, \ldots, q^{\mathrm{n_{dof}}}) \in \mathbb{R}^{\mathrm{n_{dof}}} \subset \mathbb{R}^{3N},\ \ n_{\mathrm{dof}} = 3N - k \right\} \tag{2.35}$$

Thus, we have $\mathbf{q} = (q^1, ..., q^{n_{\mathrm{dof}}}) \in Q \subset \mathbb{R}^{n_{\mathrm{dof}}=3N-k}$; a set of generalized coordinates in the n_{dof}-dimensional configuration manifold Q. At every point $\mathbf{q}$ on the differentiable manifold Q, a set of generalized velocities $\dot{\mathbf{q}} = (\dot{q}^1, ..., \dot{q}^{\mathrm{n_{dof}}}) \in T_q Q \subset \mathbb{R}^{3N-k}$ defines an $\mathrm{n_{dof}}$-dimensional tangent space $T_q Q$. Note that $\dim Q = \dim T_q Q$. The tangent bundle, sometimes called as the phase velocity space, is a disjoint union of the tangent spaces to each point of the configuration manifold Q as

$$TQ := \left\{ (\mathbf{q}, \dot{\mathbf{q}}) \mid \mathbf{q} \in Q,\ \dot{\mathbf{q}} \in T_q Q \right\} = \bigcup_{q \in Q} T_q Q \tag{2.36}$$

By arranging Newton's equation of motion, i.e., Newton's second law; see Eq. (2.27), D'Alembert's principle is obtained:

$$\sum_{a=1}^{N} \mathbf{f}_a^{\mathrm{appl}} + \sum_{a=1}^{N} \mathbf{C}_a + \left(- \sum_{a=1}^{N} m_a \frac{d^2 \mathbf{x}^a(t)}{dt^2} \right) = \mathbf{0} \tag{2.37}$$

where the last term of the left-hand side of equation right above is called an inertial force or a reversed effective force. Suppose the constraint conditions are smooth; namely, the constraint force $\mathbf{C}_a$ is orthogonal to the configuration manifold and satisfies

$$\sum_{a=1}^{N} \mathbf{C}_a \cdot \delta \mathbf{x}^a = 0 \tag{2.38}$$

In other words, the virtual work done by the constraint force is zero since the constraint force is workless. Note that Eq. (2.38) is true in both rheonomic and scleronomic systems. Notice that Eq. (2.38) is not always true if we use the exact differential of $\mathbf{x}^a$, i.e., $d\mathbf{x}^a$, instead of the variation $\delta \mathbf{x}^a$ even if $\mathbf{C}_a$ is orthogonal to the configuration manifold: In the rheonomic system, we have

$$\sum_{a=1}^{N} \mathbf{C}_a \cdot d\mathbf{x}^a = \sum_{a=1}^{N} \left(\mathbf{C}_a \cdot \frac{\partial \mathbf{x}^a}{\partial q^i} \right) dq^i + \sum_{a=1}^{N} \left(\mathbf{C}_a \cdot \frac{\partial \mathbf{x}^a}{\partial t} \right) dt \tag{2.39}$$

although, in the scleronomics system, we have

$$\sum_{a=1}^{N} \mathbf{C}_a \cdot d\mathbf{x}^a = \sum_{a=1}^{N} \mathbf{C}_a \cdot \delta\mathbf{x}^a = 0 \tag{2.40}$$

In view of Eq. (2.38), the Lagrangian form of D'Alembert's principle in generalized coordinates may be written as

$$\sum_{a=1}^{N} \left[\left(\mathbf{f}_a^{\text{appl}} - m_a \frac{d^2\mathbf{x}^a}{dt^2} \right) \cdot \frac{\partial \mathbf{x}^a}{\partial q^i} \right] \delta q^i = 0 \tag{2.41}$$

Note $\delta\mathbf{q}$ are linearly independent and aribitrary; therefore, we have

$$\sum_{a=1}^{N} \left[\left(\mathbf{f}_a^{\text{appl}} - m_a \frac{d^2\mathbf{x}^a}{dt^2} \right) \cdot \frac{\partial \mathbf{x}^a}{\partial q^i} \right] = 0 \tag{2.42}$$

Defining the notion of generalized force for the applied force as

$$Q_i^{\text{appl}} := \sum_{a=1}^{N} \mathbf{f}_a^{\text{appl}} \cdot \frac{\partial \mathbf{x}^a}{\partial q^i} \tag{2.43}$$

then, Eq. (2.42) can be written as

$$\sum_{a=1}^{N} m_a \frac{d^2\mathbf{x}^a}{dt^2} \cdot \frac{\partial \mathbf{x}^a}{\partial q^i} = Q_i^{\text{appl}} \tag{2.44}$$

Rearranging the left-hand side of Eq. (2.44) yields

$$\begin{aligned} \text{LHS} &= \sum_{a=1}^{N} m_a \frac{d^2\mathbf{x}^a}{dt^2} \cdot \frac{\partial \mathbf{x}^a}{\partial q^i} \\ &= \sum_{a=1}^{N} \left[\frac{d}{dt} \left(m_a \dot{\mathbf{x}}^a \cdot \frac{\partial \mathbf{x}^a}{\partial q^i} \right) - m_a \dot{\mathbf{x}}^a \cdot \frac{d}{dt} \left(\frac{\partial \mathbf{x}^a}{\partial q^i} \right) \right] \\ &= \frac{d}{dt} \left(\sum_{a=1}^{N} m_a \dot{\mathbf{x}}^a \cdot \frac{\partial \dot{\mathbf{x}}^a}{\partial \dot{q}^i} \right) - \sum_{a=1}^{N} m_a \dot{\mathbf{x}}^a \cdot \frac{\partial \dot{\mathbf{x}}^a}{\partial q^i} \end{aligned} \tag{2.45}$$

Notice that we have used the following relations during the above calculations:

$$\dot{\mathbf{x}}^a = \frac{\partial \mathbf{x}^a}{\partial t} + \frac{\partial \mathbf{x}^a}{\partial q^i} \dot{q}^i \tag{2.46}$$

$$\frac{\partial \dot{\mathbf{x}}^a}{\partial \dot{q}^i} = \frac{\partial \mathbf{x}^a}{\partial q^i} \tag{2.47}$$

$$\frac{d}{dt} \left(\frac{\partial \mathbf{x}^a}{\partial q^i} \right) = \frac{\partial}{\partial q^i} \left(\frac{\partial \mathbf{x}^a}{\partial q^j} \right) \dot{q}^j + \frac{\partial}{\partial q^i} \left(\frac{\partial \mathbf{x}^a}{\partial t} \right) = \frac{\partial \dot{\mathbf{x}}^a}{\partial q^i} \tag{2.48}$$

Equation (2.47) is called the dot cancellation. For the N-body dynamical system undergoing a rectilinear motion, the kinetic energy $\mathscr{T}(\dot{\mathbf{x}}) : \mathbb{R}^{3N} \to \mathbb{R}$ in Cartesian coordinates is identical to the kinetic energy $\mathscr{T}(\mathbf{q}, \dot{\mathbf{q}}, t) : TQ \times \mathbb{I} \to \mathbb{R}$ in generalized coordinates, i.e.,

$$\begin{aligned}
\mathscr{T}(\dot{\mathbf{x}}(\mathbf{q}, t)) &= \mathscr{T}(\mathbf{q}, \dot{\mathbf{q}}, t) \\
&= \frac{1}{2}\sum_{a=1}^{N} m_a \dot{\mathbf{x}}^a(\mathbf{q}, t) \cdot \dot{\mathbf{x}}^a(\mathbf{q}, t) \\
&= \frac{1}{2}\sum_{a=1}^{N} m_a \frac{\partial \mathbf{x}^a}{\partial q^i} \cdot \frac{\partial \mathbf{x}^a}{\partial q^j}\dot{q}^i\dot{q}^j + \sum_{a=1}^{N} m_a \frac{\partial \mathbf{x}^a}{\partial q^i} \cdot \frac{\partial \mathbf{x}^a}{\partial t}\dot{q}^i + \frac{1}{2}\sum_{a=1}^{N} m_a \frac{\partial \mathbf{x}^a}{\partial t} \cdot \frac{\partial \mathbf{x}^a}{\partial t} \\
&= \underbrace{\frac{1}{2} m_{ij}\dot{q}^i\dot{q}^j}_{\mathscr{T}_2} + \underbrace{\sum_{a=1}^{N} m_a \frac{\partial \mathbf{x}^a}{\partial q^i} \cdot \frac{\partial \mathbf{x}^a}{\partial t}\dot{q}^i}_{\mathscr{T}_1} + \underbrace{\frac{1}{2}\sum_{a=1}^{N} m_a \frac{\partial \mathbf{x}^a}{\partial t} \cdot \frac{\partial \mathbf{x}^a}{\partial t}}_{\mathscr{T}_0}
\end{aligned} \tag{2.49}$$

where $\mathscr{T}_2$ is a quadratic function of generalized velocities, $\mathscr{T}_1$ is a linear function of generalized velocities, and $\mathscr{T}_0$ is a function of generalized coordinates and time. For the holonomic-scleronomic constraint system, the kinetic energy, which is a quadratic function of $\dot{\mathbf{x}}$, is also a quadratic function of $\dot{\mathbf{q}}$, because $\mathscr{T}_1 = \mathscr{T}_0 = 0$ due to $\frac{\partial \mathbf{x}}{\partial t} = 0$. Thus, the kinetic energy in the holonomic-scleronomic constraint system can be given by $\mathscr{T}(\dot{\mathbf{x}}) = \mathscr{T}_2(\mathbf{q}, \dot{\mathbf{q}}) = \mathscr{T}(\mathbf{q}, \dot{\mathbf{q}})$. However, in the holonomic-rheonomic constraint system; $\mathscr{T}_1 \neq 0$ and $\mathscr{T}_0 \neq 0$. Note that the symmetric quasi-mass $m_{ij} = m_{ji}$ is defined by

$$m_{ij} := \sum_{a=1}^{N} m_a \frac{\partial \mathbf{x}^a}{\partial q^i} \cdot \frac{\partial \mathbf{x}^a}{\partial q^j} \tag{2.50}$$

and is a function of generalized coordinates. The kinetic energy is the translational kinetic energy rather than the rotational kinetic energy. Our attention is restricted to the translational kinetic energy only of the dynamical system because the rotation effects are neglected in this derivation. Then, the variation of the kinetic energy can be written as

$$\delta\mathscr{T}(\dot{\mathbf{x}}) = \sum_{a=1}^{N} m_a \dot{\mathbf{x}}^a \cdot \delta\dot{\mathbf{x}}^a = \sum_{a=1}^{N} m_a \dot{\mathbf{x}}^a \cdot \left(\frac{\partial \dot{\mathbf{x}}^a}{\partial q^i}\delta q^i + \frac{\partial \dot{\mathbf{x}}^a}{\partial \dot{q}^i}\delta\dot{q}^i \right) \tag{2.51}$$

Also, the variation of the kinetic energy can be given by

$$\delta\mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}},t\right)=\frac{\partial\mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial q^{i}}\delta q^{i}+\frac{\partial\mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial\dot{q}^{i}}\delta\dot{q}^{i} \tag{2.52}$$

In view of Eq. (2.51) and Eq. (2.52), we have the relations

$$\sum_{a=1}^{N}m_{a}\dot{\mathbf{x}}^{a}\cdot\frac{\partial\dot{\mathbf{x}}^{a}}{\partial q^{i}}=\frac{\partial\mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial q^{i}}\ \text{ and }\ \sum_{a=1}^{N}m_{a}\dot{\mathbf{x}}^{a}\cdot\frac{\partial\dot{\mathbf{x}}^{a}}{\partial\dot{q}^{i}}=\frac{\partial\mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial\dot{q}^{i}} \tag{2.53}$$

Hence, Eq. (2.45) becomes

$$\text{LHS}=\frac{d}{dt}\left(\frac{\partial\mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial\dot{q}^{i}}\right)-\frac{\partial\mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial q^{i}} \tag{2.54}$$

In the special case where the conservative force is given from the potential energy, i.e.,

$$Q_{i}^{\text{appl}}=Q_{i}^{\text{c}}+Q_{i}^{\text{nc}}$$

where the generalized forces for the conservative and non-conservative forces are given as

$$Q_{i}^{\text{c}}=\sum_{a=1}^{N}\mathbf{f}_{a}^{\text{c}}\cdot\frac{\partial\mathbf{x}^{a}}{\partial q^{i}}=-\frac{\partial\mathscr{U}\left(\mathbf{q},t\right)}{\partial q^{i}} \tag{2.55}$$

$$Q_{i}^{\text{nc}}=\sum_{a=1}^{N}\mathbf{f}_{a}^{\text{nc}}\cdot\frac{\partial\mathbf{x}^{a}}{\partial q^{i}} \tag{2.56}$$

respectively, Eq. (2.44) in the indicial notation yields

$$\begin{aligned}&\frac{d}{dt}\left(\frac{\partial\mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial\dot{q}^{i}}\right)-\frac{\partial\mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial q^{i}}=-\frac{\partial\mathscr{U}\left(\mathbf{q},t\right)}{\partial q^{i}}+Q_{i}^{\text{nc}}\\ \Leftrightarrow\quad&\frac{d}{dt}\left(\frac{\partial\mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial\dot{q}^{i}}\right)-\frac{\partial\left(\mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}},t\right)-\mathscr{U}\left(\mathbf{q},t\right)\right)}{\partial q^{i}}=Q_{i}^{\text{nc}}\end{aligned} \tag{2.57}$$

Next, we introduce a scalar function called the Lagrangian $\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right):TQ\times\mathbb{I}\rightarrow\mathbb{R}$ which is defined as

$$\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right):=\mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}},t\right)-\mathscr{U}\left(\mathbf{q},t\right) \tag{2.58}$$

Therefore, Eq. (2.57) can be written as

$$\frac{d}{dt}\left(\frac{\partial \mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial \dot{q}^i}\right)-\frac{\partial \mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial q^i}=Q_i^{\mathrm{nc}} \tag{2.59}$$

for $i=1,2,\ldots,n_{\mathrm{dof}}$. Equation (2.59) is called Lagrange's equations of motion and indicates n_{dof}-simultaneous differential equations of the system. In the vector-matrix notation,

$$\boxed{\frac{d}{dt}\left(\frac{\partial \mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial \dot{\mathbf{q}}}\right)-\frac{\partial \mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial \mathbf{q}}=\mathbf{Q}^{\mathrm{nc}}} \tag{2.60}$$

In absence of the generalized force for the non-conservative force, i.e., $\mathbf{Q}^{\mathrm{nc}}=\mathbf{0}$, Lagrange's equation of motion yields:

$$\boxed{\frac{d}{dt}\left(\frac{\partial \mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial \dot{\mathbf{q}}}\right)-\frac{\partial \mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial \mathbf{q}}=\mathbf{0}} \tag{2.61}$$

Rearranging Lagrange's equation of motion, see Eq. (2.61), yields

$$\frac{\partial^2 \mathscr{L}}{\partial \dot{q}^i \partial \dot{q}^j}\frac{d^2 q^j}{dt^2}+\frac{\partial^2 \mathscr{L}}{\partial \dot{q}^i \partial q^j}\frac{dq^j}{dt}+\frac{\partial^2 \mathscr{L}}{\partial \dot{q}^i \partial t}-\frac{\partial \mathscr{L}}{\partial q^i}=0 \tag{2.62}$$

for $i=1,2,\ldots,n_{\mathrm{dof}}$. Suppose the Lagrangian $\mathscr{L}$ is *regular* (or *strongly nondegenerate*); namely, the Hessian matrix of the Lagrangian defined by $\mathbf{G}:=\frac{\partial^2 \mathscr{L}(\mathbf{q},\dot{\mathbf{q}},t)}{\partial \dot{q}^i \partial \dot{q}^j}$ satisfies

$$\det\left[\mathbf{G}\right]=\det\left[\frac{\partial^2 \mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial \dot{q}^i \partial \dot{q}^j}\right]\neq 0 \tag{2.63}$$

In this case, Lagrange's equation of motion can be written by

$$\frac{d^2 q^j}{dt^2}=-\left(\mathbf{G}^{-1}\right)^{ij}\left(\frac{\partial^2 \mathscr{L}}{\partial \dot{q}^i \partial q^k}\frac{dq^k}{dt}+\frac{\partial^2 \mathscr{L}}{\partial \dot{q}^i \partial t}-\frac{\partial \mathscr{L}}{\partial q^i}\right) \tag{2.64}$$

and we can see that Lagrange's equation is a second-order ordinary differential equation in time t.

In honolonomic system, substituting Eq. (2.49) into the standard form of Lagrange's equation of motion, i.e., Eq. (2.59) with $Q_i^{\mathrm{nc}}=0$, we obtain the

following expression:

$$m_{ij}\ddot{q}^j + \frac{1}{2}\left(\frac{\partial m_{ij}}{\partial q^k} + \frac{\partial m_{ik}}{\partial q^j} - \frac{\partial m_{jk}}{\partial q^i}\right)\dot{q}^j\dot{q}^k + \left(\frac{\partial m_{ij}}{\partial t} + \frac{\partial a^i}{\partial q^j} - \frac{\partial \alpha^j}{\partial q^i}\right)\dot{q}^j + \frac{\partial \alpha^i}{\partial q^j} - \frac{\partial \mathscr{T}_0}{\partial q^i} + \frac{\partial \mathscr{U}}{\partial q^i} = 0 \tag{2.65}$$

where α^i is defined as

$$\alpha^i := \sum_{a=1}^{N} m_a \frac{\partial \mathbf{x}^a}{\partial q^i} \cdot \frac{\partial \mathbf{x}^a}{\partial t} \tag{2.66}$$

and m_{ij} and α^i depends upon the generalized coordinates and time. In an autonomous system, i.e., the holonomic-scleronomic system, the kinetic energy is given as a homogeneous quadratic function of $\dot{\mathbf{q}}$; therefore, substituting $\mathscr{T} = \mathscr{T}_2 = \frac{1}{2}m_{ij}\dot{q}^i\dot{q}^j$ into Lagrange's equations of motion, we obtain

$$m_{ij}\ddot{q}^j + \frac{1}{2}\left(\frac{\partial m_{ij}}{\partial q^k} + \frac{\partial m_{ik}}{\partial q^j} - \frac{\partial m_{jk}}{\partial q^i}\right)\dot{q}^j\dot{q}^k + \frac{\partial \mathscr{U}}{\partial q^i} = 0 \tag{2.67}$$

This system is also called as the natural system; see [6].

Total Energy Representation of Lagrange's Equation of Motion

Besides the Lagrangian $\mathscr{L}(\mathbf{q}, \dot{\mathbf{q}}, t) : TQ \times \mathbb{I} \to \mathbb{R}$, we herein introduce a new scalar descriptive function, namely, the Total Energy $\mathscr{E}(\mathbf{q}, \dot{\mathbf{q}}, t) : TQ \times \mathbb{I} \to \mathbb{R}$ as an alternative. The generalized coordinates on the trajectory curve still belong to the configuration manifold Q. In the configuration manifold, the dynamical system for the Total Energy is not subject to any constraints as in the Lagrangian dynamical system. Note that the configuration manifold is an n_{dof}-dimensional smooth manifold, and it is also differentiable. Consider the holonomic system. We define the Total Energy as the summation of the kinetic energy $\mathscr{T} : TQ \times \mathbb{I} \to \mathbb{R}$ and the potential energy $\mathscr{U}(\mathbf{q}, t) : Q \times \mathbb{I} \to \mathbb{R}$ as

$$\mathscr{E}(\mathbf{q}, \dot{\mathbf{q}}, t) := \mathscr{T}(\mathbf{q}, \dot{\mathbf{q}}, t) + \mathscr{U}(\mathbf{q}, t) \tag{2.68}$$

In view of Eq. (2.57), we obtain the following equation ($Q_i^{\text{nc}} = 0$):

$$\boxed{\frac{d}{dt}\left(\frac{\partial \mathscr{E}(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial \dot{q}^i}\right) + \frac{\partial \mathscr{E}(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial q^i} - 2\frac{\partial \mathscr{T}(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial q^i} = 0} \tag{2.69}$$

Equation (2.69) is the equation of motion for the holonomic system in the total energy representation and framework. We simply call it the *total energy representation of the equation of motion*. When the kinetic energy does not depend upon the generalized coordinates, i.e., when the symmetric generalized mass, m_{ij}, is constant, Eq. (2.69) can be written by

$$\frac{d}{dt}\left(\frac{\partial \mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial \dot{q}^i}\right)+\frac{\partial \mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial q^i}=0 \tag{2.70}$$

In the autonomous total energy system, i.e., the holonomic-scleronomic system, the kinetic energy is given as a homogeneous quadratic function of $\dot{\mathbf{q}}$; therefore, substituting $\mathscr{T}=\mathscr{T}_2=\frac{1}{2}m_{ij}\dot{q}^i\dot{q}^j$ into the total energy representation of the equation of motion, we obtain

$$m_{ij}\ddot{q}^j+\frac{1}{2}\left(\frac{\partial m_{ij}}{\partial q^k}+\frac{\partial m_{ik}}{\partial q^j}-\frac{\partial m_{jk}}{\partial q^i}\right)\dot{q}^j\dot{q}^k+\frac{\partial \mathscr{U}}{\partial q^i}=0 \tag{2.71}$$

This system is also called as the natural system; see [6].

2.1.3 Hamilton's Equations of Motion (Canonical Equations)

Further, in the Lagrangian framework, one can define the *generalized momentum* p_i by

$$p_i\left(\mathbf{q},\dot{\mathbf{q}},t\right):=\frac{\partial \mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial \dot{q}^i} \tag{2.72}$$

for $i=1,2,\ldots,n_{\mathrm{dof}}$, which is associated with the generalized velocities $\dot{q}^i$. It should be noted that if q^i is a Cartesian coordinate variable, then the corresponding generalized momentum p_i becomes the linear momentum; however, if q^i is a curvilinear coordinate, then the corresponding generalized momentum p_i becomes the angular momentum. The generalized force for the conservative force Q_i^c can be written with the generalized momentum by an indicial notation as

$$Q_i^c=-\frac{\partial \mathscr{U}\left(\mathbf{q},t\right)}{\partial q^i}=\frac{dp_i}{dt}-\frac{\partial \mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial q^i} \tag{2.73}$$

Therefore, Eq. (2.61) can be expressed as the following system of equations:

$$\frac{dp_i\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{dt} = \frac{\partial\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial q^i} \tag{2.74a}$$

$$p_i\left(\mathbf{q},\dot{\mathbf{q}},t\right) = \frac{\partial\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial\dot{q}^i} \tag{2.74b}$$

for $i = 1, 2, \ldots, n_{\text{dof}}$, or in the vector-matrix notation,

$$\frac{d\mathbf{p}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{dt} = \frac{\partial\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial\mathbf{q}} \tag{2.75a}$$

$$\mathbf{p}\left(\mathbf{q},\dot{\mathbf{q}},t\right) = \frac{\partial\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial\dot{\mathbf{q}}} \tag{2.75b}$$

Using the generalized momentum, consider the variation of the Lagrangian:

$$\begin{aligned}
\delta\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right) &= \frac{\partial\mathscr{L}}{\partial q^i}\delta q^i + \frac{\partial\mathscr{L}}{\partial\dot{q}^i}\delta\dot{q}^i + \frac{\partial\mathscr{L}}{\partial t}\delta t \\
&= \frac{\partial\mathscr{L}}{\partial q^i}\delta q^i + p_i\delta\dot{q}^i + \frac{\partial\mathscr{L}}{\partial t}\delta t \\
&= \frac{\partial\mathscr{L}}{\partial q^i}\delta q^i + \delta\left(p_i\dot{q}^i\right) - \dot{q}^i\delta p_i + \frac{\partial\mathscr{L}}{\partial t}\delta t \\
\Leftrightarrow \delta\left(p_i\dot{q}^i - \mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right)\right) &= -\frac{\partial\mathscr{L}}{\partial q^i}\delta q^i + \dot{q}^i\delta p_i - \frac{\partial\mathscr{L}}{\partial t}\delta t
\end{aligned} \tag{2.76}$$

Now, define the *Jacobi integral* $h\left(\mathbf{q},\dot{\mathbf{q}},t\right)$ by

$$h\left(\mathbf{q},\dot{\mathbf{q}},t\right) := \underbrace{\frac{\partial\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial\dot{q}^i}}_{p_i}\dot{q}^i - \mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right) = \mathscr{H}\left(\mathbf{q},\mathbf{p}\left(\mathbf{q},\dot{\mathbf{q}},t\right),t\right) \tag{2.77}$$

such that

$$\delta\mathscr{H}\left(\mathbf{q},\mathbf{p}\left(\mathbf{q},\dot{\mathbf{q}},t\right),t\right) = -\frac{\partial\mathscr{L}}{\partial q^i}\delta q^i + \dot{q}^i\delta p_i - \frac{\partial\mathscr{L}}{\partial t}\delta t \tag{2.78}$$

It should be noted that Eq. (2.78) shows Eq. (2.77) is valid for any $(\mathbf{q},\dot{\mathbf{q}}) \in TQ$ if the generalized momentum is given by Eq. (2.72). In order to express Eq. (2.74)

in terms of the Hamiltonian $\mathscr{H}$, consider Eq. (2.74b) firstly, i.e.,

$$\begin{aligned} p_i\left(\mathbf{q},\dot{\mathbf{q}},t\right) &= \frac{\partial\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial\dot{q}^i} = \frac{\partial\left[p_i\dot{q}^i-\mathscr{H}\left(\mathbf{q},\mathbf{p}\left(\mathbf{q},\dot{\mathbf{q}},t\right),t\right)\right]}{\partial\dot{q}^i} \\ &= p_i+\dot{q}^j\frac{\partial p_j}{\partial\dot{q}^i}-\frac{\partial\mathscr{H}}{\partial p_j}\frac{\partial p_j}{\partial\dot{q}^i} \\ &= p_i+\left(\dot{q}^j-\frac{\partial\mathscr{H}}{\partial p_j}\right)\frac{\partial p_j}{\partial\dot{q}^i} \\ &= p_i+\left(\dot{q}^j-\frac{\partial\mathscr{H}}{\partial p_j}\right)\frac{\partial^2\mathscr{L}}{\partial\dot{q}^i\partial\dot{q}^j} \end{aligned} \tag{2.79}$$

Therefore, if the Lagrangian is regular, i.e., if Eq. (2.63) is satisfied, we obtain

$$\dot{q}^i = \frac{\partial\mathscr{H}}{\partial p_i} \tag{2.80}$$

Secondly, consider Eq. (2.74a):

$$\begin{aligned} \frac{dp_i\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{dt} &= \left.\frac{\partial\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right)}{\partial q^i}\right|_{\dot{\mathbf{q}}} = \left.\frac{\partial\left[p_i\dot{q}^i-\mathscr{H}\left(\mathbf{q},\mathbf{p}\left(\mathbf{q},\dot{\mathbf{q}},t\right),t\right)\right]}{\partial q^i}\right|_{\dot{\mathbf{q}}} \\ &= \dot{q}^j\frac{\partial p_j}{\partial q^i}-\left.\frac{\partial\mathscr{H}}{\partial q^i}\right|_{\mathbf{p}}-\frac{\partial\mathscr{H}}{\partial p_j}\frac{\partial p_j}{\partial q^i} \\ &= -\left.\frac{\partial\mathscr{H}}{\partial q^i}\right|_{\mathbf{p}}+\left(\dot{q}^j-\frac{\partial\mathscr{H}}{\partial p_j}\right)\frac{\partial p_j}{\partial q^i} \\ &= -\left.\frac{\partial\mathscr{H}}{\partial q^i}\right|_{\mathbf{p}}+\left(\dot{q}^j-\frac{\partial\mathscr{H}}{\partial p_j}\right)\frac{\partial^2\mathscr{L}}{\partial q^i\partial\dot{q}^j} \end{aligned} \tag{2.81}$$

Substituting Eq. (2.80), we obtain

$$\frac{dp_i}{dt} = -\left.\frac{\partial\mathscr{H}}{\partial q^i}\right|_{\mathbf{p}} \quad\text{and}\quad \left.\frac{\partial\mathscr{L}}{\partial q^i}\right|_{\dot{\mathbf{q}}} = -\left.\frac{\partial\mathscr{H}}{\partial q^i}\right|_{\mathbf{p}} \tag{2.82}$$

It should be noted that we can solve Eq. (2.72) uniquely if Eq. (2.63) is satisfied according to Eq. (2.80);

$$\dot{q}^i = \psi^i\left(\mathbf{q},\mathbf{p},t\right) \tag{2.83}$$

Hence, via the transformation

$$\begin{aligned} &\left(\mathbf{q},\dot{\mathbf{q}}\right)\in TQ \longmapsto \left(\mathbf{q},\mathbf{p}\right)\in T^*Q; \\ &\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right)\longmapsto \mathscr{H}\left(\mathbf{q},\mathbf{p},t\right) = p_i\psi^i-\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}},t\right) \end{aligned} \tag{2.84}$$

where T^*Q is the cotangent bundle defined by

$$T^*Q := \left\{ (\mathbf{q}, \mathbf{p}) \mid \mathbf{q} \in Q,\ \dot{\mathbf{q}} \in T_q^*Q \right\} = \bigcup_{q \in Q} T_q^*Q \tag{2.85}$$

the following Hamilton's equations of motion are given:

$$\dot{p}_i = -\frac{\partial \mathscr{H}(\mathbf{q}, \mathbf{p}, t)}{\partial q^i} \tag{2.86a}$$

$$\dot{q}^i = \frac{\partial \mathscr{H}(\mathbf{q}, \mathbf{p}, t)}{\partial p_i} \tag{2.86b}$$

for $i = 1, 2, \ldots, n_{\text{dof}}$, or in the vector-matrix notation,

$$\dot{\mathbf{p}} = -\frac{\partial \mathscr{H}(\mathbf{q}, \mathbf{p}, t)}{\partial \mathbf{q}} \tag{2.87a}$$

$$\dot{\mathbf{q}} = \frac{\partial \mathscr{H}(\mathbf{q}, \mathbf{p}, t)}{\partial \mathbf{p}} \tag{2.87b}$$

The transformation above is called the Legendre transformation. In physics, the cotangent bundle is frequently called the *phase space* P, i.e., $P \equiv T^*Q$, and p_i is now called the *canonical momentum*. The generalized velocity $\dot{\mathbf{q}} = (\dot{q}^1, ..., \dot{q}^{\text{n}_{\text{dof}}}) \in T_qQ$ is a vector-valued function as a tangent vector on the tangent space to the configuration manifold at every point $\dot{\mathbf{q}}$. In view of Eq. (2.72), it is implied that the space having the canonical momenta $\mathbf{p} = (p_1, ..., p_{\text{n}_{\text{dof}}}) \in T_q^*Q$ as local coordinates is dual to the tangent space. The dual space to the tangent space T_qQ is called the cotangent space T_q^*Q. Hence, the canonical momenta are dual vectors, or covectors, or covariant vectors, or one-forms on the cotangent space. The disjoint union of the cotangent and configuration manifold is called the cotangent bundle T^*Q. Note $\dim T_qQ = \dim T_q^*Q = n_{\text{dof}}$ and $2\left(\dim T_q^*Q\right) = \dim TQ$.

2.2 Scalar Formalisms: Equations of Motion for a System of N Particles in Newtonian, Lagrangian, and Hamiltonian Mechanics-Variational Calculus Setting

2.2.1 Hamilton's Principle and Modified Hamilton's Principle

It was W.R. Hamilton that firstly stated that the equations of motion are equivalent to the variational principle. The variational principle is known as Hamilton's principle, which is sometimes called the principle of stationary action; see Principle 2. It is an integral principle, and it can be postulated as a fundamental principle of the mechanics, in fact, it is cited to be more general than Newton's equations of motion which have a limitation [7].

Theorem 2 (Hamilton's Principle)
Consider the Lagrangian $\mathscr{L} : TQ \to \mathbb{R}$. *The actual curve (path)* $\mathbf{c}_0 : [t_0, t_1] \subset \mathbb{R} \to Q : t \longmapsto \mathbf{c}_0(t)$ *on the configuration manifold* Q, *that is,* $\mathbf{c}_0(t) = \mathbf{q}(t) = (\dot{q}^1, \dot{q}^2, \ldots, \dot{q}^{\mathrm{n_{dof}}})$, *joining two points* $\mathbf{q}_A(t_0)$ *and* $\mathbf{q}_B(t_1)$ *at fixed time is given when the action integral (functional) defined by*

$$S[\mathbf{q}] := \int_{t_1}^{t_2} \mathscr{L}(\mathbf{q}(t), \dot{\mathbf{q}}(t), t)\, dt = 0 \tag{2.88a}$$

is stationary with fixed endpoints.

Consider a Lagrangian flow with generalized coordinates; that is, we assume the solution of a system of N particles is a point on the configuration manifold.

$$\left.\frac{dS}{d\varepsilon}\right|_{\varepsilon=0} := \left.\frac{d}{d\varepsilon}\right|_{\varepsilon=0} \int_{t_1}^{t_2} \mathscr{L}(\mathbf{q}(t), \dot{\mathbf{q}}(t), t)\, dt = 0 \tag{2.89}$$

or we may write the above equation with the variation as

$$\delta S := \delta \int_{t_1}^{t_2} \mathscr{L}(\mathbf{q}(t), \dot{\mathbf{q}}(t), t)\, dt = 0 \tag{2.90}$$

with the boundary conditions: $\delta\mathbf{q}(t_1) = \mathbf{0}$ and $\delta\mathbf{q}(t_2) = \mathbf{0}$. The varied action integral can be represented as

$$S[\mathbf{q} + \varepsilon\boldsymbol{\eta}] = \int_{t_1}^{t_2} \mathscr{L}(\mathbf{q} + \varepsilon\boldsymbol{\eta}, \dot{\mathbf{q}} + \varepsilon\dot{\boldsymbol{\eta}}, t)\, dt = 0 \tag{2.91}$$

where $\boldsymbol{\eta}(t) = (\eta^1(t), \eta^2(t), \ldots, \eta^{\mathrm{n_{dof}}}(t))$ denote the admissible functions satisfying the boundary conditions: $\boldsymbol{\eta}(t_0) = \mathbf{0}$ and $\boldsymbol{\eta}(t_1) = \mathbf{0}$. Note that $\boldsymbol{\eta}$ represents the direction of the virtual displacement. In order for the varied action to be stationary at $\varepsilon = 0$, the Gateaux variation of the action integral must vanish as follows:

$$\left.\frac{d}{d\varepsilon}\right|_{\varepsilon=0} S[\mathbf{q} + \varepsilon\boldsymbol{\eta}] = 0 \tag{2.92}$$

In other words, the action integral on the actual curve is stationary when Eq. (2.92) is satisfied. From (2.92), we obtain

$$\int_{t_1}^{t_2} \left[\frac{\partial\mathscr{L}(\mathbf{q} + \varepsilon\boldsymbol{\eta}, \dot{\mathbf{q}} + \varepsilon\dot{\boldsymbol{\eta}}, t)}{\partial(q^i + \varepsilon\eta^i)}\eta^i + \frac{\partial\mathscr{L}(\mathbf{q} + \varepsilon\boldsymbol{\eta}, \dot{\mathbf{q}} + \varepsilon\dot{\boldsymbol{\eta}}, t)}{\partial(\dot{q}^i + \varepsilon\dot{\eta}^i)}\dot{\eta}^i\right]\Bigg|_{\varepsilon=0} dt = 0 \tag{2.93}$$

for $i = 1, 2, \ldots, n_{\mathrm{dof}}$, and it yields

$$\int_{t_1}^{t_2} \left[\frac{\partial\mathscr{L}(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial q^i} - \frac{d}{dt}\left(\frac{\partial\mathscr{L}(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial\dot{q}^i}\right)\right]\eta^i dt + \left.\frac{\partial\mathscr{L}(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial\dot{q}^i}\eta^i\right|_{t=t_0}^{t=t_1} = 0 \tag{2.94}$$

Since the admissible functions satisfy the boundary conditions, the second term in Eq. (2.94) vanishes; hence,

$$\int_{t_1}^{t_2} \left[\frac{\partial\mathscr{L}(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial q^i} - \frac{d}{dt}\left(\frac{\partial\mathscr{L}(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial\dot{q}^i}\right)\right]\eta^i dt = 0 \tag{2.95}$$

For the arbitrariness of the admissible functions within the time interval, the Euler-Lagrange equation is readily obtained:

$$\frac{d}{dt}\left(\frac{\partial\mathscr{L}(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial\dot{q}^i}\right) - \frac{\partial\mathscr{L}(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial q^i} = 0 \tag{2.96}$$

Consequently, Hamilton's principle may be rewritten as follows: The actual curve (path) $\mathbf{q}_0 : [t_1, t_2] \subset \mathbb{R} \to Q : t \longmapsto \mathbf{q}_0(t)$ on the configuration manifold Q with

fixed endpoints at fixed time is given if and only if the Euler-Lagrange equation is satisfied. For the autonomous Lagrangian system, the above may be written as

$$\begin{aligned} \delta S &:= \delta \int_{t_1}^{t_2} \mathscr{L}\left(\mathbf{q}(t), \dot{\mathbf{q}}(t)\right) dt \\ &= \int_{t_1}^{t_2} \left[\frac{\partial \mathscr{L}\left(\mathbf{q}, \dot{\mathbf{q}}\right)}{\partial q^i} - \frac{d}{dt}\left(\frac{\partial \mathscr{L}\left(\mathbf{q}, \dot{\mathbf{q}}\right)}{\partial \dot{q}^i} \right) \right] \delta q^i dt = 0 \end{aligned} \tag{2.97}$$

if and only if

$$\boxed{\frac{\partial \mathscr{L}\left(\mathbf{q}, \dot{\mathbf{q}}\right)}{\partial q^i} - \frac{d}{dt}\left(\frac{\partial \mathscr{L}\left(\mathbf{q}, \dot{\mathbf{q}}\right)}{\partial \dot{q}^i} \right) = 0} \tag{2.98}$$

with the boundary conditions: $\delta\mathbf{q}(t_0) = \varepsilon\eta(t_0) = \mathbf{0}$ and $\delta\mathbf{q}(t_1) = \varepsilon\eta(t_1) = \mathbf{0}$. Alternatively, for the autonomous Lagrangian system, after obtaining the Euler-Lagrange equation, the Total Energy representation of the equation of motion is readily obtained via (which is not a Legendre transformation)

$$\mathscr{L}\left(\mathbf{q}, \dot{\mathbf{q}}\right) = \frac{\partial \mathscr{E}\left(\mathbf{q}, \dot{\mathbf{q}}\right)}{\partial \dot{q}^i} \dot{q}^i - \mathscr{E}\left(\mathbf{q}, \dot{\mathbf{q}}\right) \tag{2.99}$$

That is,

$$\boxed{\frac{d}{dt}\left(\frac{\partial \mathscr{E}\left(\mathbf{q}, \dot{\mathbf{q}}\right)}{\partial \dot{q}^i} \right) + \frac{\partial \mathscr{E}\left(\mathbf{q}, \dot{\mathbf{q}}\right)}{\partial q^i} - 2\frac{\partial \mathscr{T}\left(\mathbf{q}, \dot{\mathbf{q}}\right)}{\partial q^i} = 0} \tag{2.100}$$

However, invoking the Legendre transform, i.e., Eq. (2.84), the Hamilton's equations of motion for the autonomous Lagrangian/Hamiltonian system can be readily obtained from the Euler-Lagrange equation; that is,

$$\boxed{\dot{p}_i = -\frac{\partial \mathscr{H}\left(\mathbf{q}, \mathbf{p}\right)}{\partial q^i} \;\text{ and }\; \dot{q}^i = \frac{\partial \mathscr{H}\left(\mathbf{q}, \mathbf{p}\right)}{\partial p_i}} \tag{2.101}$$

Note that Eq. (2.99) is true only for the autonomous Lagrangian/Total Energy system whereas the Legendre transform, i.e., Eq. (2.84), is true even for the nonautonomous Lagrangian/Hamiltonian system.

Deriving the Euler-Lagrange equation of motion directly from Hamilton's principle, i.e., $\delta \int_{t_1}^{t_2} \mathscr{L} dt = 0$, and then deriving Hamilton's canonical equations via the Legendre transformation is a natural way of the derivation of equations of motion

from Hamilton's principle. And, the total energy representation of the equation of motion can be readily obtained via Eq. (2.99) from the Euler-Lagrange equation of motion. As the colloraries of Hamilton's principle, we may also substitute Eq. (2.99) or Eq. (2.84) at the first step as

$$\delta \int_{t_1}^{t_2} \left[\frac{\partial \mathscr{E}\left(\mathbf{q}, \dot{\mathbf{q}}\right)}{\partial \dot{q}^i} \dot{q}^i - \mathscr{E}\left(\mathbf{q}, \dot{\mathbf{q}}\right) \right] dt = 0 \tag{2.102}$$

$$\delta \int_{t_1}^{t_2} \left[p_i \dot{q}^i - \mathscr{H}\left(\mathbf{q}, \mathbf{p}\right) \right] dt = 0 \tag{2.103}$$

for the derivation of Eq. (2.100) and Eq. (2.101), respectively.

Next, consider Eq. (2.102). Assume that the autonomous total energy is defined as $\mathscr{E}\left(\mathbf{q}, \dot{\mathbf{q}}\right) := \mathscr{T}\left(\mathbf{q}, \dot{\mathbf{q}}\right) + \mathscr{U}\left(\mathbf{q}\right)$ where the kinetic energy is given by $\mathscr{T}\left(\mathbf{q}, \dot{\mathbf{q}}\right) = \frac{1}{2} m_{ij}(\mathbf{q}) \dot{q}^i \dot{q}^j$. Therefore, Eq. (2.102) may be written as

$$\delta S = \delta \int_{t_1}^{t_2} \left[m_{ij}\left(\mathbf{q}\right) \dot{q}^i \dot{q}^j - \mathscr{E}\left(\mathbf{q}, \dot{\mathbf{q}}\right) \right] dt = 0 \tag{2.104}$$

Carrying out the variation, we have

$$\begin{aligned} & 2 m_{ij}\left(\mathbf{q}\right) \dot{q}^i \delta q^j \Big|_{t=t_0}^{t=t_1} - \frac{\partial \mathscr{E}}{\partial \dot{q}^i} \delta q^i \Big|_{t=t_0}^{t=t_1} \\ & \quad - \int_{t_1}^{t_2} \left[\frac{d}{dt} \left(\frac{\partial \mathscr{E}\left(\mathbf{q}, \dot{\mathbf{q}}\right)}{\partial \dot{q}^i} \right) + \frac{\partial \mathscr{E}\left(\mathbf{q}, \dot{\mathbf{q}}\right)}{\partial q^i} - 2 \frac{\partial \mathscr{T}\left(\mathbf{q}, \dot{\mathbf{q}}\right)}{\partial q^i} \right] \delta q^i dt = 0 \end{aligned} \tag{2.105}$$

Since the endpoints are fixed at fixed time, the above equation becomes

$$\int_{t_1}^{t_2} \left[\frac{d}{dt} \left(\frac{\partial \mathscr{E}\left(\mathbf{q}, \dot{\mathbf{q}}\right)}{\partial \dot{q}^i} \right) + \frac{\partial \mathscr{E}\left(\mathbf{q}, \dot{\mathbf{q}}\right)}{\partial q^i} - 2 \frac{\partial \mathscr{T}\left(\mathbf{q}, \dot{\mathbf{q}}\right)}{\partial q^i} \right] \delta q^i dt = 0 \tag{2.106}$$

Therefore, for the satisfaction of $\delta S = 0$, the autonomous total energy representation of the equation of motion must be satisfied. Hamilton's principle for the autonomous total energy framework may be summarized as follows: The actual curve (path) $\mathbf{c}_0 : [t_1, t_2] \subset \mathbb{R} \to Q : t \longmapsto \mathbf{c}_0(t)$ on the configuration manifold Q with fixed endpoints $\delta\mathbf{q}(t_1) = \delta\mathbf{q}(t_2) = \mathbf{0}$ at fixed time is given if and only if the autonomous total energy representation of the equation of motion is satisfied.

Next, consider Hamilton's principle on the cotangent bundle T^*Q, i.e., Eq. (2.103). Carrying out the variation, we get

$$p_i \delta q^i \big|_{t=t_0}^{t=t_1} - \int_{t_1}^{t_2} \left[\left(\dot{p}_i - \frac{\partial \mathscr{H}}{\partial q^i} \right) \delta q^i + \left(\dot{q}^i - \frac{\partial \mathscr{H}}{\partial p_i} \right) \delta p_i \right] dt = 0 \tag{2.107}$$

Since the end points in the configuration manifold Q are fixed, we have $\delta \mathbf{q}(t_1) = \delta \mathbf{q}(t_2) = \mathbf{0}$; therefore, the above equation may reduce to

$$\int_{t_1}^{t_2} \left[\left(\dot{p}_i - \frac{\partial \mathscr{H}}{\partial q^i} \right) \delta q^i + \left(\dot{q}^i - \frac{\partial \mathscr{H}}{\partial p_i} \right) \delta p_i \right] dt = 0 \tag{2.108}$$

The actual curve $\mathbf{c}_0$ on the cotangent bundle is the curve connecting the points in $D_1 \subset T^*Q$ and $D_2 \subset T^*Q$; for Hamilton's principle on the cotangent bundle, we cannot choose an arbitrary endpoint independently from a given starting point as can we for Hamilton's principle on the configuration manifold. Because δq^i and δp_i are not independent of each other, their coefficients in Eq. (2.108) must be zero for $\delta S = 0$; that is,

$$\dot{p^i} = -\frac{\partial \mathscr{H}\left(\mathbf{q}, \mathbf{p}\right)}{\partial q^i} \text{ and } \dot{q}^i = \frac{\partial \mathscr{H}\left(\mathbf{q}, \mathbf{p}\right)}{\partial p_i} \tag{2.109}$$

Hence, Hamilton's principle on the cotangent bundle may be written as follows: The actual curve (path) $\mathbf{c}_0 : [t_1, t_2] \subset \mathbb{R} \to T^*Q : t \longmapsto \mathbf{c}_0(t)$ on the cotangent bundle T^*Q with fixed endpoints $\delta \mathbf{q}(t_1) = \delta \mathbf{q}(t_2) = \mathbf{0}$ at fixed time is given if and only if Hamilton's equations of motion are satisfied. Notice $\delta \mathbf{p}(t_1) = \delta \mathbf{p}(t_2) = \mathbf{0}$ is not required in general. Hamilton's principle on the cotangent bundle is sometimes called the modified Hamilton's principle; see [6]. The summary of Hamilton's principles and its relations to the equations of motion in the autonomous Lagrangian/Hamiltonian/total energy frameworks are shown in Figure 2.1. Note that it is the Euler-Lagrange equation that is first derived directly from Hamilton's principle; then, the Hamilton's equations of motion are derived via the Legendre transform, or the total energy representation of the equation of motion is derived by substitutions from the Euler-Lagrange equation (see [7, 8] for other details). It is suggested that the use of Hamilton's law of varying action, which is not a

variational principle, as an alternative to the Hamilton's principle due to the fact that the latter has some logical inconsistencies; see [7, 8] .

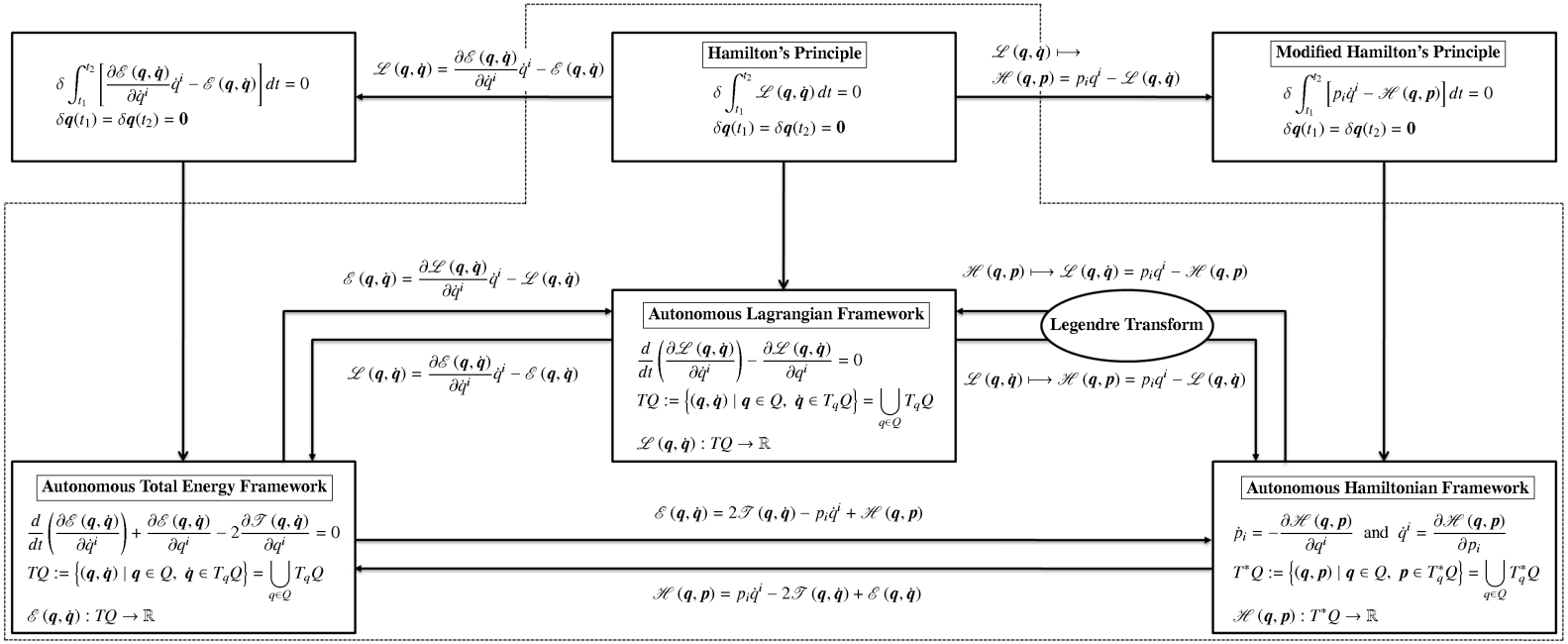


Figure 2.1: The illustration of Hamilton's principle and the relations to the equations of motion in the scalar formalisms.

2.2.2 Principle of Balance of Mechanical Energy - Differential Calculus Setting

The focus in this subsection is on a dynamical system restricted to be holonomic-scleronomic. Consider an autonomous total mechanical energy in a system and the setting of differential calculus, i.e., an autonomous total energy in a mechanical system, defined by

$$\mathscr{E}(\mathbf{q},\dot{\mathbf{q}}) := \frac{\partial \mathscr{L}(\mathbf{q},\dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}} \cdot \dot{\mathbf{q}} - \mathscr{L}(\mathbf{q},\dot{\mathbf{q}}) \tag{2.110}$$

where $\mathbf{q}(t) : \mathbb{I} = [t_0, t_f] \subset \mathbb{R} \rightarrow Q$ denotes the generalized coordinates on a smooth differentiable n_{dof}-dimensional configuration manifold Q, and $\dot{\mathbf{q}}(t) = d\mathbf{q}(t)/dt : \mathbb{I} \rightarrow T_{\mathbf{q}}Q$ denotes the system velocity on the tangent space at $\mathbf{q} \in Q$; and $\mathscr{L}(\mathbf{q},\boldsymbol{\nu}) : TQ \rightarrow \mathbb{R}$ denotes the autonomous Lagrangian given by the kinetic energy, $\mathscr{T}(\mathbf{q},\dot{\mathbf{q}}) : TQ \rightarrow \mathbb{R}$, minus the potential energy, $\mathscr{U}(\mathbf{q}) : Q \rightarrow \mathbb{R}$, of the system, i.e.,

$$\mathscr{L}(\mathbf{q},\dot{\mathbf{q}}) = \mathscr{T}(\mathbf{q},\dot{\mathbf{q}}) - \mathscr{U}(\mathbf{q}) \tag{2.111}$$

The Total Energy defined in Eq. (2.110) is equivalent to that the kinetic plus the potential energy of the system, i.e.,

$$\mathscr{E}(\mathbf{q},\dot{\mathbf{q}}) = \mathscr{T}(\mathbf{q},\dot{\mathbf{q}}) + \mathscr{U}(\mathbf{q}) \tag{2.112}$$

If the kinetic energy is given homogeneous quadratic form as

$$\mathscr{T}(\mathbf{q},\dot{\mathbf{q}}) = \frac{1}{2}\dot{\mathbf{q}} \cdot \mathbf{M}(\mathbf{q})\dot{\mathbf{q}} \tag{2.113}$$

with a positive definite, symmetric mass matrix $\mathbf{M}$ due to Euler's theorem $(\partial \mathscr{T}(\mathbf{q},\dot{\mathbf{q}})/\partial \dot{\mathbf{q}}) \cdot \dot{\mathbf{q}} = 2\mathscr{T}(\mathbf{q},\dot{\mathbf{q}})$. Note that we assume that the total potential energy depends only upon the generalized coordinates: $\mathscr{U}(\mathbf{q}) : Q \rightarrow \mathbb{R}$.

Theorem 3 (Principle of Balance of Mechanical Energy)
The balance of energy in mechanical systems is given by,

$$\boxed{\frac{d\mathscr{E}(\mathbf{q},\dot{\mathbf{q}})}{dt} = \mathscr{P}_{\text{input}} \;\; \forall t \in \mathbb{I} = [t_0, t_f]} \tag{2.114a}$$

where the mechanical energy and $\mathscr{P}_{\text{input}} \in \mathbb{R}$ *denotes the total power input due to generalized nonconservative forces,* $\mathbf{Q}^{\text{nc}}(t)$*, i.e.,*

$$\mathscr{P}_{\text{input}} = \mathbf{Q}^{\text{nc}}(t) \cdot \dot{\mathbf{q}} \tag{2.114b}$$

which yields the Total Energy representation of the equation of motion as

$$\frac{d}{dt}\left(\frac{\partial \mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial \dot{\mathbf{q}}}\right) + \frac{\partial \mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial \mathbf{q}} - 2\frac{\partial \mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial \mathbf{q}} = \mathbf{Q}^{\text{nc}} \tag{2.114c}$$

Proof. Eq. (2.114a) yields

$$\begin{aligned} 0 &= \frac{d\mathscr{E}(\mathbf{q},\dot{\mathbf{q}})}{dt} - \mathscr{P}_{\text{input}} = \frac{\partial \mathscr{E}}{\partial \dot{\mathbf{q}}} \cdot \ddot{\mathbf{q}} + \frac{\partial \mathscr{E}}{\partial \mathbf{q}} \cdot \dot{\mathbf{q}} - \mathbf{Q}^{\text{nc}} \cdot \dot{\mathbf{q}} \\ &= \frac{\partial \mathscr{T}}{\partial \dot{\mathbf{q}}} \cdot \ddot{\mathbf{q}} + \left[\frac{\partial \mathscr{E}}{\partial \mathbf{q}} - \mathbf{Q}^{\text{nc}}\right] \cdot \dot{\mathbf{q}} \end{aligned} \tag{2.115}$$

Here, we have the following relation:

$$\begin{aligned} \frac{\partial \mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial \dot{\mathbf{q}}} \cdot \ddot{\mathbf{q}} &= \mathbf{M}(\mathbf{q})\dot{\mathbf{q}} \cdot \ddot{\mathbf{q}} = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} \cdot \dot{\mathbf{q}} \\ &= \left[\frac{d(\mathbf{M}(\mathbf{q})\dot{\mathbf{q}})}{dt} - \dot{\mathbf{M}}(\mathbf{q})\dot{\mathbf{q}}\right] \cdot \dot{\mathbf{q}} \\ &= \left[\frac{d}{dt}\left(\frac{\partial \mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial \dot{\mathbf{q}}}\right) - 2\frac{\partial}{\partial \mathbf{q}}\left(\frac{1}{2}\dot{\mathbf{q}} \cdot \mathbf{M}(\mathbf{q})\dot{\mathbf{q}}\right)\right] \cdot \dot{\mathbf{q}} \\ &= \left[\frac{d}{dt}\left(\frac{\partial \mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial \dot{\mathbf{q}}}\right) - 2\frac{\partial \mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial \mathbf{q}}\right] \cdot \dot{\mathbf{q}} \end{aligned} \tag{2.116}$$

due to $\mathbf{M}^T(\mathbf{q}) = \mathbf{M}(\mathbf{q})$. Notice that

$$\mathbf{M}\ddot{\mathbf{q}} = \frac{d}{dt}\left(\frac{\partial \mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial \dot{\mathbf{q}}}\right) \tag{2.117}$$

if $\mathbf{M}$ is constant. Thus,

$$\left[\frac{d}{dt}\left(\frac{\partial \mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial \dot{\mathbf{q}}}\right) + \frac{\partial \mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial \mathbf{q}} - 2\frac{\partial \mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial \mathbf{q}} - \mathbf{Q}^{\text{nc}}\right] \cdot \dot{\mathbf{q}} = 0 \tag{2.118}$$

which readily gives

$$\frac{d}{dt}\left(\frac{\partial \mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial \dot{\mathbf{q}}}\right) + \frac{\partial \mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial \mathbf{q}} - 2\frac{\partial \mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial \mathbf{q}} = \mathbf{Q}^{\text{nc}} \tag{2.119}$$

because of the property of linear independence of $\dot{\mathbf{q}}(t)$. Notice that the mechanical energy defined by Eq. (2.110) can be also used to derive Eq. (2.114c) from Eq. (2.114a). That is, Eq. (2.114a) yields

$$\begin{aligned}
0 &= \frac{d\mathscr{E}(\mathbf{q},\dot{\mathbf{q}})}{dt} - \mathscr{P}_{\text{input}} \\
&= \frac{d}{dt}\left[\frac{\partial \mathscr{L}(\mathbf{q},\dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}}\cdot\dot{\mathbf{q}} - \mathscr{L}(\mathbf{q},\dot{\mathbf{q}})\right] - \mathbf{Q}^{\text{nc}}\cdot\dot{\mathbf{q}} \\
&= \frac{d}{dt}\left(\frac{\partial \mathscr{L}(\mathbf{q},\dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}}\right)\cdot\dot{\mathbf{q}} + \frac{\partial \mathscr{L}(\mathbf{q},\dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}}\cdot\ddot{\mathbf{q}} - \frac{\partial \mathscr{L}(\mathbf{q},\dot{\mathbf{q}})}{\partial \mathbf{q}}\cdot\dot{\mathbf{q}} \\
&\quad -\frac{\partial \mathscr{L}(\mathbf{q},\dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}}\cdot\ddot{\mathbf{q}} - \mathbf{Q}^{\text{nc}}\cdot\dot{\mathbf{q}} \\
&= \left[\frac{d}{dt}\left(\frac{\partial \mathscr{E}(\mathbf{q},\dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}}\right) + \frac{\partial \mathscr{E}(\mathbf{q},\dot{\mathbf{q}})}{\partial \mathbf{q}} - 2\frac{\partial \mathscr{T}(\mathbf{q},\dot{\mathbf{q}})}{\partial \mathbf{q}} - \mathbf{Q}^{\text{nc}}\right]\cdot\dot{\mathbf{q}}
\end{aligned} \tag{2.120}$$

which also leads to Eq. (2.114c) because of the property of linear independence of $\dot{\mathbf{q}}(t)$. ■

Remark 1

1. *The Total Energy representation of the equation of motion as given by Eq.* (2.114c) *in mechanical systems is, of course, equivalent to the Lagrange's equation of motion represented in terms of the Lagrangian,* $\mathscr{L}(\mathbf{q},\dot{\mathbf{q}})$*, of the system. If we assume that the kinetic energy does not depend on* $\mathbf{q}$ *in Eq.* (2.113)*, i.e., when* $\mathbf{M}$ *is constant, Eq.* (2.114c) *can be written as*

$$\frac{d}{dt}\left(\frac{\partial \mathscr{E}\,(\mathbf{q},\dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}}\right) + \frac{\partial \mathscr{E}\,(\mathbf{q},\dot{\mathbf{q}})}{\partial \mathbf{q}} = \mathbf{Q}^{\text{nc}} \tag{2.121}$$

   That is, Eq. (2.114a) *implies that we restrict the constraint to be holonomic-scleronomic.*

2. *If the dissipative potential energy is given as* $\vartheta(\mathbf{q},\dot{\mathbf{q}}) : TQ \to R$*, the principle of balance mechanical energy in the dissipative system may be written as*

$$\frac{d\mathscr{E}(\mathbf{q},\dot{\mathbf{q}})}{dt} = -\dot{\mathbf{q}}\cdot\frac{\partial \vartheta}{\partial \dot{\mathbf{q}}} \tag{2.122}$$

In linear dynamical systems, the potential energy and the dissipative potential energy may be given as

$$\mathscr{U}(\mathbf{q}) = \frac{1}{2}\mathbf{q}\cdot\mathbf{K}\mathbf{q} \tag{2.123}$$

$$\vartheta(\mathbf{q},\dot{\mathbf{q}}) = \frac{1}{2}\dot{\mathbf{q}}\cdot\mathbf{C}\dot{\mathbf{q}} \tag{2.124}$$

where $\mathbf{K}$ *is a semi-positive definite, symmetric matrix, and* $\mathbf{C}$ *is a positive definite, symmetric matrix. Therefore, Eq.* (2.122) *leads to the equation of motion which takes the form*

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{K}\mathbf{q} = -\mathbf{C}\dot{\mathbf{q}} \tag{2.125}$$

The right-hand side of the above equation is the generalized dissipative force, $\mathbf{Q}^{\text{diss}} = -\partial\vartheta/\partial\dot{\mathbf{q}} = -\mathbf{C}\dot{\mathbf{q}}$.

3. *The Total Energy representation of the equation of motion, i.e., Eq.* (2.114c), *is a second-order differential equation in* $\mathbf{q} \in Q$, *similar to the Lagrange's equation of motion. By introducing the canonical momentum*

$$\mathbf{p} := \partial\mathscr{L}(\mathbf{q},\dot{\mathbf{q}})/\partial\dot{\mathbf{q}} \tag{2.126}$$

with the transformation $(\mathbf{q},\dot{\mathbf{q}}) \in TQ \mapsto (\mathbf{q},\mathbf{p}) \in T^*Q$, *the Total Energy representation of the equation of motion can be shown to be equivalent to Hamilton's equations of motion,*

$$\dot{\mathbf{q}} = \frac{\partial\mathscr{H}(\mathbf{q},\mathbf{p})}{\partial\mathbf{p}} \text{ and } \dot{\mathbf{p}} = -\frac{\partial\mathscr{H}(\mathbf{q},\mathbf{p})}{\partial\mathbf{q}} \tag{2.127}$$

Eq. (2.127) *is a* ***first-order*** *system in* $T^*_{\mathbf{q}}Q$.

4. *In contrast to Hamilton's equations of motion, the Total Energy representation of the equation of motion is a* ***second-order*** *system in* Q, *and it can be written as*

$$\frac{\partial^2\mathscr{E}(\mathbf{q},\dot{\mathbf{q}})}{\partial\dot{\mathbf{q}}\partial\dot{\mathbf{q}}}\cdot\ddot{\mathbf{q}} + \frac{\partial^2\mathscr{E}(\mathbf{q},\dot{\mathbf{q}})}{\partial\dot{\mathbf{q}}\partial\mathbf{q}}\cdot\dot{\mathbf{q}} + \frac{\partial\mathscr{E}(\mathbf{q},\dot{\mathbf{q}})}{\partial\mathbf{q}} - \frac{\partial\mathscr{T}(\mathbf{q},\dot{\mathbf{q}})}{\partial\mathbf{q}} = \mathbf{Q}^{\text{nc}} \tag{2.128}$$

Therefore, if

$$\det\left[\frac{\partial^2 \mathscr{E}(\mathbf{q},\dot{\mathbf{q}})}{\partial\dot{\mathbf{q}}\partial\dot{\mathbf{q}}}\right] \neq 0 \tag{2.129}$$

we have

$$\ddot{\mathbf{q}} = \left[\frac{\partial^2 \mathscr{E}(\mathbf{q},\dot{\mathbf{q}})}{\partial\dot{\mathbf{q}}\partial\dot{\mathbf{q}}}\right]^{-1} \cdot \left(\frac{\partial \mathscr{T}(\mathbf{q},\dot{\mathbf{q}})}{\partial\mathbf{q}} - \frac{\partial \mathscr{E}(\mathbf{q},\dot{\mathbf{q}})}{\partial\mathbf{q}} - \frac{\partial^2 \mathscr{E}(\mathbf{q},\dot{\mathbf{q}})}{\partial\dot{\mathbf{q}}\partial\mathbf{q}} \cdot \dot{\mathbf{q}}\right) \tag{2.130}$$

5. *For a dynamical* ***nonconstrained*** *system of N particles moving in Euclidean three-space $\mathbb{R}^3$, we have $\mathbf{q}(t) = \mathbf{x}(t) \in Q \equiv \mathbb{R}^{3N} := \mathbb{R}^3 \times \cdots \times \mathbb{R}^3$ (N times) and $\mathbf{q}^i(t) = \mathbf{x}^i(t) \in \mathbb{R}^3$ for $i = 1, 2, \cdots, N$. Hence, Eq.* (2.114c) *becomes*

$$\frac{d}{dt}\left(\frac{\partial \mathscr{E}\,(\mathbf{q},\dot{\mathbf{q}})}{\partial\dot{\mathbf{q}}}\right) + \frac{\partial \mathscr{E}\,(\mathbf{q},\dot{\mathbf{q}})}{\partial\mathbf{q}} = \mathbf{Q}^{\mathrm{nc}} \tag{2.131}$$

with the nonconservative force,

$$\mathbf{Q}^{\mathrm{nc}} = \mathbf{f}^{\mathrm{nc}} \cdot \frac{\partial\mathbf{x}}{\partial\mathbf{q}} = \mathbf{f}^{\mathrm{nc}} \in \mathbb{R}^{3N} \tag{2.132}$$

which is equivalent to Newton's equation of motion,

$$\mathbf{M}\ddot{\mathbf{x}} + \mathbf{f}^{\mathrm{int}} = \mathbf{f}^{\mathrm{c}} + \mathbf{f}^{\mathrm{nc}} \tag{2.133}$$

where $\mathbf{M}$ denotes the $3N \times 3N$ constant mass matrix given by

$$\mathbf{M} = \mathrm{diag}(\mathbf{m}_1, \mathbf{m}_2, \cdots, \mathbf{m}_N) \tag{2.134}$$

with $\mathbf{m}_i = m_i\mathbf{I}_3$ where $\mathbf{I}_3$ is the identity matrix for $i = 1, 2, \cdots, N$; $\mathbf{f}^{\mathrm{int}} \in \mathbb{R}^{3N}$ and $\mathbf{f}^{\mathrm{c}} \in \mathbb{R}^{3N}$ denote the internal force due to interaction among particles, and the external conservative force defined by the internal and external potential energy, respectively, as

$$\mathbf{f}^{\mathrm{int}}(\mathbf{x}) := \frac{\partial \mathscr{U}^{\mathrm{int}}(\mathbf{x})}{\partial\mathbf{x}} \;\; \textit{and} \;\; \mathbf{f}^{\mathrm{c}}(\mathbf{x}) := -\frac{\partial \mathscr{U}^{\mathrm{ext}}(\mathbf{x})}{\partial\mathbf{x}} \tag{2.135}$$

Here, we have assumed the (total) potential energy, $\mathscr{U}(\mathbf{x}) : \mathbb{R}^{3N} \to \mathbb{R}$, is given as the summation of the internal and external potential energy of the system as $\mathscr{U} = \mathscr{U}^{\mathrm{int}} + \mathscr{U}^{\mathrm{ext}}$.

Corollary 2 (Principle of Conservation of Total Mechanical Energy) *If $\mathscr{P}_{\text{input}} = 0$ in Theorem 3, the conservation of the Total Energy in mechanical systems,*

$$\boxed{\frac{d\mathscr{E}(\mathbf{q}(t), \dot{\mathbf{q}}(t))}{dt} = 0 \;\; \forall t \in \mathbb{I} = [t_0, t_f]} \tag{2.136}$$

leads to the Total Energy representation of the equation of motion in the single-field form representation in conservative systems as

$$\frac{d}{dt}\left(\frac{\partial \mathscr{E}(\mathbf{q}, \dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}}\right) + \frac{\partial \mathscr{E}(\mathbf{q}, \dot{\mathbf{q}})}{\partial \mathbf{q}} = 2\frac{\partial \mathscr{T}(\mathbf{q}, \dot{\mathbf{q}})}{\partial \mathbf{q}} \tag{2.137}$$

If the kinetic energy does not depend on $\mathbf{q}$*, we have*

$$\frac{d}{dt}\left(\frac{\partial \mathscr{E}(\mathbf{q}, \dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}}\right) + \frac{\partial \mathscr{E}(\mathbf{q}, \dot{\mathbf{q}})}{\partial \mathbf{q}} = \mathbf{0} \tag{2.138}$$

Dead Load and Nondead Load Cases

The power input, $\mathscr{P}_{\text{input}} \in \mathbb{R}$, by the generalized nonconservative external forces, $\mathbf{Q}^{\text{nc}}$, is defined as

$$\mathscr{P}_{\text{input}} := \mathbf{Q}^{\text{nc}} \cdot \dot{\mathbf{q}} = \mathbf{f}^{\text{nc}} \cdot \frac{\partial \mathbf{x}}{\partial \mathbf{q}} \cdot \dot{\mathbf{q}} \tag{2.139}$$

Assume that the nonconservative force $\mathbf{f}^{\text{nc}} \in \mathbb{R}^{3N}$ is given as the summation of dissipative force $\mathbf{f}^{\text{diss}}$ and $\mathbf{f}$; that is,

$$\mathbf{f}^{\text{nc}} = \mathbf{f}^{\text{diss}} + \mathbf{f} \tag{2.140}$$

Therefore, Eq. (2.139) can be written as

$$\begin{aligned} \mathscr{P}_{\text{input}} &= \mathscr{P}^{\text{D}}_{\text{input}} + \mathscr{P}^{\text{F}}_{\text{input}} \\ \text{where } \; \mathscr{P}^{\text{D}}_{\text{input}} &:= \mathbf{f}^{\text{diss}} \cdot \frac{\partial \mathbf{x}}{\partial \mathbf{q}} \cdot \dot{\mathbf{q}} \;\text{ and }\; \mathscr{P}^{\text{F}}_{\text{input}} := \mathbf{f} \cdot \frac{\partial \mathbf{x}}{\partial \mathbf{q}} \cdot \dot{\mathbf{q}} \end{aligned} \tag{2.141}$$

Suppose the nonconservative dissipative force is given as a function of velocities as

$$\mathbf{f}^{\text{diss}} = -\mathbf{C}\dot{\mathbf{x}} \tag{2.142}$$

The symmetric and positive-definite matrix $\mathbf{C}$ is assumed constant for simplicity ($\mathbf{C}$ can depend on velocities in general; see [9]). The generalized nonconservative dissipative force is given as

$$\mathbf{Q}^{\text{diss}} = -\mathbf{C}\dot{\mathbf{x}} \cdot \frac{\partial \mathbf{x}}{\partial \mathbf{q}} = -\mathbf{C}\dot{\mathbf{x}} \cdot \frac{\partial \dot{\mathbf{x}}}{\partial \dot{\mathbf{q}}} \tag{2.143}$$

Dead Load Case: We call the dead load case when the nonconservative external force $\mathbf{f}$ is given constant. In this case, we can define the power input by $\mathbf{f}$ as the rate of change of $\mathscr{U}^{\text{F}}(\mathbf{q}) : Q \to \mathbb{R}$; that is,

$$\mathscr{P}^{\text{F}}_{\text{input}} = \mathbf{f} \cdot \frac{\partial \mathbf{x}}{\partial \mathbf{q}} \cdot \dot{\mathbf{q}} =: \frac{d\mathscr{U}^{\text{F}}}{dt} \tag{2.144}$$

and the total potential energy is now given as

$$\mathscr{U}(\mathbf{q}) := \mathscr{U}^{\text{int}}(\mathbf{q}) + \mathscr{U}^{\text{ext}}(\mathbf{q}) = \mathscr{U}^{\text{int}}(\mathbf{q}) + \mathscr{U}^{\text{c}}(\mathbf{q}) + \mathscr{U}^{\text{F}}(\mathbf{q}) \tag{2.145}$$

where $\mathscr{U}^{\text{int}}(\mathbf{q})$ and $\mathscr{U}^{\text{c}}(\mathbf{q})$ denote the internal potential energy related with an internal force and the external potential energy related with an external conservative force, respectively. Also, we define the external potential energy, $\mathscr{U}^{\text{ext}}(\mathbf{q})$, as the summation of $\mathscr{U}^{\text{c}}(\mathbf{q})$ and $\mathscr{U}^{\text{F}}(\mathbf{q})$ in this case. Note the generalized external potential force is given as

$$\mathbf{Q}^{\text{epot}} = \mathbf{Q}^{\text{c}} + \mathbf{Q}^{\text{F}} \tag{2.146}$$

where the generalized external conservative force and the generalized time independent force are defined as

$$Q^{\text{c}}_i := \mathbf{f}^{\text{c}} \cdot \frac{\partial \mathbf{x}}{\partial \mathbf{q}} := -\frac{\partial \mathscr{U}^{\text{c}}}{\partial \mathbf{q}} \tag{2.147}$$

$$Q^{\text{F}}_i := \mathbf{f} \cdot \frac{\partial \mathbf{x}}{\partial \mathbf{q}} := -\frac{\partial \mathscr{U}^{\text{F}}}{\partial \mathbf{q}} \tag{2.148}$$

Also, the external potential energy for $\mathbf{Q}^{\text{F}}$ is given as

$$\mathscr{U}^{\text{F}}(\mathbf{q}) = \mathbf{q} \cdot \mathbf{Q}^{\text{F}} \tag{2.149}$$

In the dead load case, the principle of balance of total energy in the mechanical system is written as

$$\frac{d\mathscr{E}(\mathbf{q},\dot{\mathbf{q}})}{dt} = \mathscr{P}^{\mathrm{D}}_{\mathrm{input}} \tag{2.150}$$

Nondead Load Case: Similarly, we recall the nondead load case when $\mathbf{f}$ is time-dependent, i.e., $\mathbf{f}(t) : \mathbb{I} \to \mathbb{R}^{3N}$. In this case, we cannot define $\mathscr{U}^{\mathrm{F}}$ and include it in the total potential energy; the total potential energy in this case is defined as $\mathscr{U}(\mathbf{q}) := \mathscr{U}^{\mathrm{int}}(\mathbf{q}) + \mathscr{U}^{\mathrm{c}}(\mathbf{q})$, i.e., $\mathscr{U}^{\mathrm{ext}}(\mathbf{q}) = \mathscr{U}^{\mathrm{c}}(\mathbf{q})$, and the principle of balance of total energy in the mechanical system is written as

$$\frac{d\mathscr{E}(\mathbf{q},\dot{\mathbf{q}})}{dt} = \mathscr{P}^{\mathrm{D}}_{\mathrm{input}} + \mathscr{P}^{\mathrm{F}}_{\mathrm{input}} \tag{2.151}$$

Derivation

Previously, we described total energy representation in the generalized coordinates. Alternatively, here, starting from Newton's equation of motion in Cartesian coordinate system, we derive the principle of balance of mechanical energy via the transformation from the Cartesian coordinate system to the generalized coordinate system which serves as the starting point. Consider a dynamical system of N-particles moving in the three-dimensional Euclidean space. A set of differential equations governing the rectilinear motion of the dynamical system in the Cartesian coordinate system is given by Newton's equations of motion, see Eq. (2.27):

$$\sum_{a=1}^{N} m_a \frac{d^2\mathbf{x}^a(t)}{dt^2} = \sum_{a=1}^{N} \mathbf{f}^{\mathrm{c}}_a + \sum_{a=1}^{N} \mathbf{f}^{\mathrm{nc}}_a + \sum_{a=1}^{N} \mathbf{C}_a : \mathbb{I} \to \mathbb{R}^{3N},\ \forall t \in \mathbb{I} = [t_0, t_1] \tag{2.152}$$

for $a = 1, 2, \ldots, N$. By moving the inertia term on the left-hand side of Eq. (2.152) to the right-hand side, D'Alembert's principle, which means that the dynamical equation is dynamically in an equilibrium state is obtained as follows:

$$\sum_{a=1}^{N} \mathbf{f}^{\mathrm{c}}_a + \sum_{a=1}^{N} \mathbf{f}^{\mathrm{nc}}_a + \sum_{a=1}^{N} \mathbf{C}_a - \sum_{a=1}^{N} m_a \frac{d^2\mathbf{x}^a}{dt^2} = \mathbf{0} \tag{2.153}$$

Premultiplying by the velocity $\dot{\mathbf{x}}^a(t) : \mathbb{I} \to \mathbb{R}^{3N}$ in place of the virtual displacement $\delta\mathbf{x}^a(t) : \mathbb{I} \to \mathbb{R}^{3N}$, we have

$$\left[\sum_{a=1}^{N} \mathbf{f}_a^{\mathrm{c}} + \sum_{a=1}^{N} \mathbf{f}_a^{\mathrm{nc}} + \sum_{a=1}^{N} \mathbf{C}_a - \sum_{a=1}^{N} m_a \frac{d^2\mathbf{x}^a}{dt^2}\right] \cdot \dot{\mathbf{x}}^a = 0 \tag{2.154}$$

Recasting Eq. (2.154) yields

$$\sum_{a=1}^{N} m_a \frac{d^2\mathbf{x}^a}{dt^2} \cdot \dot{\mathbf{x}}^a - \sum_{a=1}^{N} \mathbf{f}_a^{\mathrm{c}} \cdot \dot{\mathbf{x}}^a = \sum_{a=1}^{N} [\mathbf{f}_a^{\mathrm{nc}} + \mathbf{C}_a] \cdot \dot{\mathbf{x}}^a \tag{2.155}$$

Invoking the following relations of the kinetic energy $\mathscr{T}(\dot{\mathbf{x}}) : \mathbb{R}^{3N} \to \mathbb{R}$ and the potential energy $\mathscr{U}(\mathbf{x}) : \mathbb{R}^{3N} \to \mathbb{R}$; that is,

$$\mathscr{T}(\dot{\mathbf{x}}) = \frac{1}{2} \sum_{a=1}^{N} m_a \dot{\mathbf{x}}^a \cdot \dot{\mathbf{x}}^a \tag{2.156}$$

$$\mathbf{f}_a^{\mathrm{c}} = -\frac{\partial \mathscr{U}(\mathbf{x})}{\partial \mathbf{x}^a} \tag{2.157}$$

respectively, we have

$$\frac{d\mathscr{T}(\dot{\mathbf{x}})}{dt} = \sum_{a=1}^{N} m_a \dot{\mathbf{x}}^a \cdot \frac{d\dot{\mathbf{x}}^a}{dt} = \sum_{a=1}^{N} m_a \frac{d\dot{\mathbf{x}}^a}{dt} \cdot \dot{\mathbf{x}}^a = \sum_{a=1}^{N} m_a \frac{d^2\mathbf{x}^a}{dt^2} \cdot \dot{\mathbf{x}}^a \tag{2.158}$$

$$\frac{d\mathscr{U}(\mathbf{x})}{dt} = -\sum_{a=1}^{N} \mathbf{f}_a^{\mathrm{c}} \cdot \frac{d\mathbf{x}^a}{dt} = -\sum_{a=1}^{N} \mathbf{f}_a^{\mathrm{c}} \cdot \dot{\mathbf{x}}^a \tag{2.159}$$

Defining the power input by

$$\mathscr{P}_{\mathrm{input}} := \sum_{a=1}^{N} [\mathbf{f}_a^{\mathrm{nc}} + \mathbf{C}_a] \cdot \dot{\mathbf{x}}^a = \sum_{a=1}^{N} \mathbf{f}_a^{\mathrm{nc}} \cdot \dot{\mathbf{x}}^a \tag{2.160}$$

Eq. (2.155) can now be written as

$$\boxed{\frac{d\mathscr{E}(\mathbf{x}, \dot{\mathbf{x}})}{dt} = \frac{d\mathscr{T}(\dot{\mathbf{x}})}{dt} + \frac{d\mathscr{U}(\mathbf{x})}{dt} = \mathscr{P}_{\mathrm{input}}} \tag{2.161}$$

Equation (2.161) is called the theorem of expended power in the Cartesian coordinate system for N particles. It is very important that, to derive Eq. (2.161),

the constraints are assumed to be *scleronomic*, viz., holonomic-scleronomic or nonholonomic-scleronomic, confining the motion to the smooth surface. Then, the constraint force is orthogonal to the velocity, since the virtual displacement shares the same direction with the velocity in the scleronomic constraint systems; in other words, the constraint force does not create any power as it does not make any contribution in the virtual work; namely, if the constraint conditions are scleronomic, then $\sum_{a=1}^{N} \mathbf{C}_a \cdot \dot{\mathbf{x}}^a = 0$. Also, we assume the potential energy depends only on $\mathbf{x}$ explicitly; see Eq. (2.157). The above assumptions imply the total energy of the system is autonomous, i.e., $\mathscr{E} : \mathbb{R}^{3N} \to \mathbb{R} : (\mathbf{x}, \dot{\mathbf{x}}) \longmapsto \mathscr{E}(\mathbf{x}, \dot{\mathbf{x}})$. Note that the scalar time differential equation in Eq. (2.161) is subject to holonomic constraint equations since the differential system is expressed in the Cartesian coordinate system. To eliminate the constraints, we introduce generalized coordinates $\mathbf{q} \in Q$. As discussed in subsection 2.1.2, we define the generalized coordinates and the generalized velocities as

$$\mathbf{q} = (q^1, q^2, \ldots, q^{\mathrm{n_{dof}}}) \in Q \tag{2.162}$$

$$\dot{\mathbf{q}} = (\dot{q}^1, \dot{q}^2, \ldots, \dot{q}^{\mathrm{n_{dof}}}) \in T_q Q \tag{2.163}$$

where the number of degrees of freedom is given as $n_{dof} = 3N - k$. The velocity in the Cartesian coordinate system with scleronomic constraint conditions can be expressed as

$$\dot{\mathbf{x}}^a(\mathbf{q}(t)) = \frac{d\mathbf{x}^a(\mathbf{q}(t))}{dt} = \frac{\partial \mathbf{x}^a}{\partial q^i}\frac{dq^i}{dt} = \frac{\partial \mathbf{x}^a}{\partial q^i}\dot{q}^i \tag{2.164}$$

Then, Eq. (2.154) can be written as

$$\left[\sum_{a=1}^{N} \mathbf{f}_a^{\mathrm{c}} + \sum_{a=1}^{N} \mathbf{f}_a^{\mathrm{nc}} + \sum_{a=1}^{N} \mathbf{C}_a - \sum_{a=1}^{N} m_a \frac{d^2\mathbf{x}^a}{dt^2}\right] \cdot \frac{\partial \mathbf{x}^a}{\partial q^i}\dot{q}^i = 0 \tag{2.165}$$

Hence, we have

$$\sum_{a=1}^{N} m_a \frac{d^2\mathbf{x}^a}{dt^2} \cdot \frac{\partial \mathbf{x}^a}{\partial q^i}\dot{q}^i - \sum_{a=1}^{N} \mathbf{f}_a^{\mathrm{c}} \cdot \frac{\partial \mathbf{x}^a}{\partial q^i}\dot{q}^i = \sum_{a=1}^{N} [\mathbf{f}_a^{\mathrm{nc}} + \mathbf{C}_a] \cdot \frac{\partial \mathbf{x}^a}{\partial q^i}\dot{q}^i \tag{2.166}$$

In the scleronomic system, we have the the following relations; see subsection 2.1.2 for details:

$$\sum_{a=1}^{N} m_a \frac{d^2\mathbf{x}^a}{dt^2} \cdot \frac{\partial \mathbf{x}^a}{\partial q^i}\dot{q}^i = \left[\frac{d}{dt}\left(\frac{\partial \mathscr{T}(\mathbf{q},\dot{\mathbf{q}})}{\partial \dot{q}^i}\right) - \frac{\partial \mathscr{T}(\mathbf{q},\dot{\mathbf{q}})}{\partial q^i}\right]\dot{q}^i \tag{2.167}$$

$$\sum_{a=1}^{N} \mathbf{f}_a^{\mathrm{c}} \cdot \frac{\partial \mathbf{x}^a}{\partial q^i}\dot{q}^i = Q_i^c \dot{q}^i = -\frac{\partial \mathscr{U}(\mathbf{q})}{\partial q^i}\dot{q}^i \tag{2.168}$$

where Q_i^c denotes the generalized force for the conservative force. Note that we assume that the potential energy depends only upon the generalized coordinates explicitly, i.e., $\mathscr{U} : Q \to \mathbb{R} : \mathbf{q} \longmapsto \mathscr{U}(\mathbf{q})$. Using the following power input in the scleronomic system

$$\mathscr{P}_{\text{input}} := \sum_{a=1}^{N} [\mathbf{f}_a^{\mathrm{nc}} + \mathbf{C}_a] \cdot \frac{\partial \mathbf{x}^a}{\partial q^i}\dot{q}^i = \sum_{a=1}^{N} \mathbf{f}_a^{\mathrm{nc}} \cdot \frac{\partial \mathbf{x}^a}{\partial q^i}\dot{q}^i \tag{2.169}$$

with Eq. (2.167) and Eq. (2.168), Eq. (2.166) becomes

$$\left[\frac{d}{dt}\left(\frac{\partial \mathscr{T}(\mathbf{q},\dot{\mathbf{q}})}{\partial \dot{q}^i}\right) - \frac{\partial \mathscr{T}(\mathbf{q},\dot{\mathbf{q}})}{\partial q^i} + \frac{\partial \mathscr{U}(\mathbf{q})}{\partial q^i}\right]\dot{q}^i = \mathscr{P}_{\text{input}} \tag{2.170}$$

In terms of the autonomous total energy $\mathscr{E}(\mathbf{q},\dot{\mathbf{q}}) := \mathscr{T}(\mathbf{q},\dot{\mathbf{q}}) + \mathscr{U}(\mathbf{q})$, Eq. (2.170) can be written as

$$\left[\frac{d}{dt}\left(\frac{\partial \mathscr{E}(\mathbf{q},\dot{\mathbf{q}})}{\partial \dot{q}^i}\right) + \frac{\partial \mathscr{E}(\mathbf{q},\dot{\mathbf{q}})}{\partial q^i} - 2\frac{\partial \mathscr{T}(\mathbf{q},\dot{\mathbf{q}})}{\partial q^i}\right]\dot{q}^i = \mathscr{P}_{\text{input}} \tag{2.171}$$

Or in terms of the autonomous Lagrangian $\mathscr{L}(\mathbf{q},\dot{\mathbf{q}}) := \mathscr{T}(\mathbf{q},\dot{\mathbf{q}}) - \mathscr{U}(\mathbf{q})$;

$$\left[\frac{d}{dt}\left(\frac{\partial \mathscr{L}(\mathbf{q},\dot{\mathbf{q}})}{\partial \dot{q}^i}\right) - \frac{\partial \mathscr{L}(\mathbf{q},\dot{\mathbf{q}})}{\partial q^i}\right]\dot{q}^i = \mathscr{P}_{\text{input}} \tag{2.172}$$

By use of the product rule and the chain rule, Eq. (2.170) can be written as

$$\frac{d}{dt}\left(\frac{\partial \mathscr{T}(\mathbf{q},\dot{\mathbf{q}})}{\partial \dot{q}^i}\dot{q}^i - \mathscr{T}(\mathbf{q},\dot{\mathbf{q}}) + \mathscr{U}(\mathbf{q})\right) = \mathscr{P}_{\text{input}} \tag{2.173}$$

In terms of the autonomous Lagrangian, Eq. (2.173) can be written as

$$\frac{d}{dt}\left(\frac{\partial \mathscr{L}(\mathbf{q},\dot{\mathbf{q}})}{\partial \dot{q}^i}\dot{q}^i - \mathscr{L}(\mathbf{q},\dot{\mathbf{q}})\right) = \mathscr{P}_{\text{input}} \tag{2.174}$$

In the scleronomic system, the kinetic energy is quadratic in the generalized velocities; therefore, Eq. (2.173) becomes

$$\frac{d\mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{dt}=\mathscr{P}_{\text{input}} \tag{2.175}$$

The scalar quantity in parenthesis in Eq. (2.174) is defined as the Hamiltonian $\tilde{\mathscr{H}}$; see Eq. (2.77). It should be noted that if the Lagrangian is autonomous, the kinetic energy is quadratic in the generalized velocities, and the potential energy depends only upon the generalized coordinates explicitly, and the Hamiltonian $\tilde{\mathscr{H}}$ becomes equivalent to the total energy of the system; namely,

$$\tilde{\mathscr{H}}\left(\mathbf{q},\mathbf{p}\left(\mathbf{q},\dot{\mathbf{q}}\right)\right):=\frac{\partial\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial\dot{q}^{i}}\dot{q}^{i}-\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}}\right)=\mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right) \tag{2.176}$$

Via the Legendre transformation, we can express Eq. (2.172) in terms of the autonomous Hamiltonian $\mathscr{H}:T^{*}Q\rightarrow\mathbb{R}:(\mathbf{q},\mathbf{p})\longmapsto\mathscr{H}(\mathbf{q},\mathbf{p})$, i.e.,

$$\frac{d\mathscr{H}\left(\mathbf{q},\mathbf{p}\right)}{dt}=\left[\dot{p}_{i}+\frac{\partial\mathscr{H}\left(\mathbf{q},\mathbf{p}\right)}{\partial q^{i}}\right]\dot{q}^{i}-\left[\dot{q}^{i}-\frac{\partial\mathscr{H}\left(\mathbf{q},\mathbf{p}\right)}{\partial p_{i}}\right]\dot{q}^{i}=\mathscr{P}_{\text{input}} \tag{2.177}$$

Summarizing, the theorem of expended power for a system of N particles in the autonomous total energy framework, the autonomous Lagrangian framewrok, and the autonomous Hamiltonian framework, respectively may be written as follows:

$$\frac{d\mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{dt}=\left[\frac{d}{dt}\left(\frac{\partial\mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial\dot{q}^{i}}\right)+\frac{\partial\mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial q^{i}}-2\frac{\partial\mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial q^{i}}\right]\dot{q}^{i}=\mathscr{P}_{\text{input}} \tag{2.178a}$$

$$\frac{d}{dt}\left(\frac{\partial\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial\dot{q}^{i}}\dot{q}^{i}-\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}}\right)\right)=\left[\frac{d}{dt}\left(\frac{\partial\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial\dot{q}^{i}}\right)-\frac{\partial\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial q^{i}}\right]\dot{q}^{i}=\mathscr{P}_{\text{input}} \tag{2.178b}$$

$$\frac{d\mathscr{H}\left(\mathbf{q},\mathbf{p}\right)}{dt}=\left[\dot{p}_{i}+\frac{\partial\mathscr{H}\left(\mathbf{q},\mathbf{p}\right)}{\partial q^{i}}\right]\dot{q}^{i}-\left[\dot{q}^{i}-\frac{\partial\mathscr{H}\left(\mathbf{q},\mathbf{p}\right)}{\partial p_{i}}\right]\dot{q}^{i}=\mathscr{P}_{\text{input}} \tag{2.178c}$$

where the power input is given as

$$\mathscr{P}_{\text{input}}=\sum_{a=1}^{N}\mathbf{f}_{a}^{\text{nc}}\cdot\frac{\partial\mathbf{x}^{a}}{\partial q^{i}}\dot{q}^{i}=Q_{i}^{\text{nc}}\dot{q}^{i} \tag{2.178d}$$

and

$$\mathscr{H}\left(\mathbf{q},\mathbf{p}\right)=\frac{\partial\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial\dot{q}^{i}}\dot{q}^{i}-\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}}\right)=\mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right):=\mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}}\right)+\mathscr{U}\left(\mathbf{q}\right) \tag{2.178e}$$

for $i=1,2,\ldots,n_{\text{dof}}$. In the conservative system, we have $\mathscr{P}_{\text{input}}=0$; therefore, Eq. (2.178a)-(2.178c) become

$$\frac{d\mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{dt}=\left[\frac{d}{dt}\left(\frac{\partial\mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial\dot{q}^{i}}\right)+\frac{\partial\mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial q^{i}}-2\frac{\partial\mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial q^{i}}\right]\dot{q}^{i}=0 \tag{2.179}$$

$$\frac{d}{dt}\left(\frac{\partial\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial\dot{q}^{i}}\dot{q}^{i}-\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}}\right)\right)=\left[\frac{d}{dt}\left(\frac{\partial\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial\dot{q}^{i}}\right)-\frac{\partial\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial q^{i}}\right]\dot{q}^{i}=0 \tag{2.180}$$

$$\frac{d\mathscr{H}\left(\mathbf{q},\mathbf{p}\right)}{dt}=\left[\dot{p}_{i}+\frac{\partial\mathscr{H}\left(\mathbf{q},\mathbf{p}\right)}{\partial q^{i}}\right]\dot{q}^{i}-\left[\dot{q}^{i}-\frac{\partial\mathscr{H}\left(\mathbf{q},\mathbf{p}\right)}{\partial p_{i}}\right]\dot{p}_{i}=0 \tag{2.181}$$

Since both of the generalized coordinates and the generalized momenta are linearly independent, if Eq. (2.179)- Eq. (2.181) are satisfied, the following equations, i.e., the equations of motion in the conservative systems, are also satisfied, respectively:

$$\frac{d}{dt}\left(\frac{\partial\mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial\dot{q}^{i}}\right)+\frac{\partial\mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial q^{i}}-2\frac{\partial\mathscr{T}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial q^{i}}=0 \tag{2.182}$$

$$\frac{d}{dt}\left(\frac{\partial\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial\dot{q}^{i}}\right)-\frac{\partial\mathscr{L}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial q^{i}}=0 \tag{2.183}$$

$$\dot{p}_{i}+\frac{\partial\mathscr{H}\left(\mathbf{q},\mathbf{p}\right)}{\partial q^{i}}=0\ \text{ and }\ \dot{q}^{i}-\frac{\partial\mathscr{H}\left(\mathbf{q},\mathbf{p}\right)}{\partial p_{i}}=0 \tag{2.184}$$

Note that in the holonomic-scleronomic system, Eq. (2.182) can be written as

$$\frac{d}{dt}\left(\frac{\partial\mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial\dot{q}^{i}}\right)+\frac{\partial\mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial q^{i}}=0 \tag{2.185}$$

Chapter 3

Equations of Motion in Continuous-Time Systems: Elastodynamical Systems

Consider a continuum body in three dimensional Euclidean space, $\mathbb{R}^3$. Let $\mathcal{B}$ denote the reference configuration of the continuum body as an open, bounded, and simply connected set with a smooth boundary open set $\partial\mathcal{B}$ in a topological space. The reference boundary is assumed to be $\partial\mathcal{B}_\sigma \cup \partial\mathcal{B}_\varphi = \partial\mathcal{B}$ with $\partial\mathcal{B}_\sigma \cap \partial\mathcal{B}_\varphi = 0$. The position of a particle in $\mathcal{B}$ is sometimes called a material point, being denoted by $\mathbf{X} \in \mathcal{B} \subset \mathbb{E}^3$. The smooth motion of the body may be described by $\mathbf{x} = \boldsymbol{\varphi}(\mathbf{X}, t) : \mathcal{B} \times \mathbb{I} \to \mathcal{S} \subset \mathbb{E}^3$ where $\mathcal{S}$ denotes the current configuration, and $\mathbb{I} = [t_0, t_N]$ with initial time $t_0 = 0$, and the final time $t_N > t_0$ denotes the time interval. A function $\boldsymbol{\varphi}(\mathbf{X}, t) \in \mathcal{S}$ is called the deformation map relative to $\mathcal{B}$, and it is assumed admissible. That is, $\mathbf{x} = \boldsymbol{\varphi}(\mathbf{X}, t) : \mathcal{B} \times \mathbb{I} \to \mathcal{S}$ is a one-to-one mapping and the deformation gradient second-order tensor field satisfies $\det(\mathrm{GRAD}\boldsymbol{\varphi}) = |\mathbf{F}| > 0$ for all $(\mathbf{X}, t) \in \mathcal{B} \times \mathbb{I}$. In the sense of functional analysis, $\boldsymbol{\varphi}$ is a dependent variable and is of class $C^{2m}(\mathcal{B})$ in space and of class $C^2(\mathbb{I})$ in

time. The smooth configuration manifold $\mathcal{C}$ can be defined as

$$\mathcal{C} = \left\{ \begin{array}{l} \boldsymbol{\varphi} : \mathcal{B} \times \mathbb{I} \to \mathcal{S} \subset \mathbb{R}^3 | \boldsymbol{\varphi} \in C^{2m}(\mathcal{B}),\ m \geq 1, \\ \boldsymbol{\varphi}_{\mathbf{X}} \in C^2(\mathbb{I}), \det(\mathrm{GRAD}\boldsymbol{\varphi}) = |\mathbf{F}| > 0, \text{ and } \boldsymbol{\varphi}|_{\partial\mathcal{B}_\varphi} = \bar{\boldsymbol{\varphi}} \end{array} \right\} \tag{3.1}$$

where $\bar{\boldsymbol{\varphi}}$ is the prescribed quantity on the prescribed displacement boundary $\partial\mathcal{B}_\varphi$. Note that GRAD denotes the gradient operator of vector fields in $\mathcal{B}$ with respect to the material point $\mathbf{X} \in \mathcal{B}$.

3.1 Vector Formalism: Strong Form of the Initial Boundary-Value Problem and Weak Form for Elastodynamical Systems - Variational Calculus Setting

This section includes: (1) the derivation of Cauchy's first equation of motion in the sense of the vector formalism, namely, from the principle of balance of total linear momentum for a continuum body, and (2) the derivation of its semi-discrete form following the Galerkin finite element method. Unlike the scalar formalisms that shall be described later, the vector formalism deals with the linear momentum vector field. More detailed discussions on the vector and scalar formalisms to derive the (governing) equations of motion can be found in [7, 8].

3.1.1 Principle of Balance of Linear Momentum

The principle of balance of linear momentum for a system in $\mathcal{S}$ can be described as

$$\frac{d}{dt}\int_{\mathcal{S}} \rho(\mathbf{x},t)\,\mathbf{v}(\mathbf{x},t)\,dV = \int_{\partial\mathcal{S}} \mathbf{t}(\mathbf{x},t,\hat{\mathbf{n}})\,dA + \int_{\mathcal{S}} \rho(\mathbf{x},t)\,\mathbf{b}(\mathbf{x},t)\,dV \tag{3.2}$$

where $\rho(\mathbf{x},t) > 0$ and $\mathbf{v}(\mathbf{x},t)$ are respectively the spatial mass density scalar field and spatial velocity vector field at $\mathbf{x} \in \mathcal{S}$ at time $t \in \mathbb{I}$; $\mathbf{t}(\mathbf{x},t,\hat{\mathbf{n}})$ is the

spatial traction vector field on $\partial\mathcal{S}$, and $\hat{\mathbf{n}}(\mathbf{x},t)$ is the outward unit normal vector field; and $\mathbf{b}(\mathbf{x},t)$ is the spatial body force vector field per unit mass. Substituting $\mathbf{x}=\boldsymbol{\varphi}(\mathbf{X},t)$, the left-hand side of Eq. (3.2) yields

$$\begin{aligned}\frac{d}{dt}\int_{\mathcal{S}}\rho(\mathbf{x},t)\,\mathbf{v}(\mathbf{x},t)\,dV &= \frac{d}{dt}\int_{\mathcal{B}}\rho(\boldsymbol{\varphi}(\mathbf{X},t),t)\,\mathbf{v}(\boldsymbol{\varphi}(\mathbf{X},t),t)\,|\mathbf{F}|\,dV\\ &= \frac{d}{dt}\int_{\mathcal{B}}\rho_0(\mathbf{X})\,\mathbf{V}(\mathbf{X},t)\,dV \qquad (3.3)\end{aligned}$$

where we have used the law of conservation of mass (continuity equation) in the Lagrangian form, i.e., $\rho(\boldsymbol{\varphi}(\mathbf{X},t),t)\,|\mathbf{F}|=\rho(\mathbf{X},t_0)\equiv\rho_0(\mathbf{X})$, and $\mathbf{V}(\mathbf{X},t)=\dot{\boldsymbol{\varphi}}(\mathbf{X},t)$ is the material velocity vector field. Recalling the Cauchy law, $\mathbf{t}(\mathbf{x},t,\hat{\mathbf{n}})=\boldsymbol{\sigma}(\mathbf{x},t)\cdot\hat{\mathbf{n}}(\mathbf{x},t)$ where $\boldsymbol{\sigma}(\mathbf{x},t)$ is the Cauchy stress tensor field in the spatial description, the right-hand side of Eq. (3.2) yields

$$\begin{aligned}&\int_{\partial\mathcal{S}}\mathbf{t}(\mathbf{x},t,\hat{\mathbf{n}})\,dA+\int_{\mathcal{S}}\rho(\mathbf{x},t)\,\mathbf{b}(\mathbf{x},t)\,dV\\ =\;&\int_{\partial\mathcal{B}}|\mathbf{F}|\,\boldsymbol{\sigma}(\boldsymbol{\varphi}(\mathbf{X},t),t)\cdot\mathbf{F}^{-T}\cdot\hat{\mathbf{n}}(\boldsymbol{\varphi}(\mathbf{X},t),t)dA\\ &+\int_{\mathcal{B}}\rho(\boldsymbol{\varphi}(\mathbf{X},t),t)\,\mathbf{b}(\boldsymbol{\varphi}(\mathbf{X},t),t)\,dV\\ =\;&\int_{\partial\mathcal{B}}\mathbf{P}(\mathbf{X},t)\cdot\hat{\mathbf{N}}(\mathbf{X},t)dA+\int_{\mathcal{B}}\rho_0(\mathbf{X})\,\mathbf{B}(\mathbf{X},t)\,dV \qquad (3.4)\end{aligned}$$

where we have used the Cauchy law via Nanson's formula, i.e., the first Piola-Kirchhoff stress tensor field is given as $\mathbf{P}(\mathbf{X},t)=|\mathbf{F}|\,\boldsymbol{\sigma}(\mathbf{X},t)\cdot\mathbf{F}^{-T}$ where $\boldsymbol{\sigma}(\mathbf{X},t)$ is the Cauchy stress tensor field in the material description; and $\mathbf{B}(\mathbf{X},t)$ is the material body force vector field per unit mass. Hence, the principle of balance of linear momentum in continuum-elastodynamics in the Lagrangian form is given as

$$\frac{d}{dt}\int_{\mathcal{B}}\rho_0(\mathbf{X})\,\dot{\boldsymbol{\varphi}}(\mathbf{X},t)\,dV=\int_{\partial\mathcal{B}}\mathbf{T}(\mathbf{X},t,\hat{\mathbf{N}})dA+\int_{\mathcal{B}}\rho_0(\mathbf{X})\,\mathbf{B}(\mathbf{X},t)\,dV \qquad (3.5)$$

where $\mathbf{T}(\mathbf{X},t,\hat{\mathbf{N}})=\mathbf{P}(\mathbf{X},t)\cdot\hat{\mathbf{N}}(\mathbf{X},t)$ is the material traction vector field on $\partial\mathcal{B}$. Invoking Reynolds transport theorem and the divergence theorem, Eq. (3.5) leads to

$$\int_{\mathcal{B}}\left[\rho_0(\mathbf{X})\,\ddot{\boldsymbol{\varphi}}-\mathrm{DIV}\,\mathbf{P}(\mathbf{X},t)-\rho_0(\mathbf{X})\,\mathbf{B}(\mathbf{X},t)\right]dV=0 \qquad (3.6)$$

Thus, the principle of balance of linear momentum in the Lagrangian form requires

$$\rho_0 \ddot{\boldsymbol{\varphi}}(\mathbf{X},t) = \mathrm{DIV}\,\mathbf{P}(\mathbf{X},t) + \rho_0 \mathbf{B}(\mathbf{X},t) \tag{3.7}$$

for $(\mathbf{X},t) \in \mathcal{B}\times\mathbb{I}$. Eq. (3.7) is called Cauchy's first law of motion [10,11] or simply Cauchy's equation of motion [12–15] and may be interpreted as the local form of balance of linear momentum. The strong form of the initial boundary-value problem for elastodynamical systems consists of Eq. (3.7), ***Cauchy's fundamental theorem*** [10, 11] (or simply ***Cauchy's law*** [12–15]) which is the prescribed traction boundary condition as the natural boundary condition, the prescribed displacement boundary condition as the essential boundary condition, plus initial conditions as follows: Find the motion map $\boldsymbol{\varphi}(\mathbf{X},t) : \mathcal{B}\times\mathbb{I} \to \mathcal{S}$, which is the dependent variable, from

$$\boxed{\begin{aligned}
\rho_0 \ddot{\boldsymbol{\varphi}}(\mathbf{X},t) &= \mathrm{DIV}\,\mathbf{P}(\mathbf{X},t) + \rho_0 \mathbf{B}(\mathbf{X},t) && \text{in } \mathcal{B}\times\mathbb{I}\\
\bar{\mathbf{T}}(\mathbf{X},t,\hat{\mathbf{N}}) &= \mathbf{P}(\mathbf{X},t)\cdot\hat{\mathbf{N}}(\mathbf{X},t) && \text{in } \partial\mathcal{B}_\sigma\times\mathbb{I}\\
\boldsymbol{\varphi} &= \bar{\boldsymbol{\varphi}} && \text{in } \partial\mathcal{B}_\varphi\times\mathbb{I}\\
\boldsymbol{\varphi}(\mathbf{X},t_0) &= \boldsymbol{\varphi}_0 = \mathbf{X} && \text{in } \mathcal{B}\\
\dot{\boldsymbol{\varphi}}(\mathbf{X},t_0) &= \dot{\boldsymbol{\varphi}}_0 = \mathbf{V}_0 && \text{in } \mathcal{B}
\end{aligned}} \tag{3.8}$$

Note that $\bar{\mathbf{T}}$, $\bar{\boldsymbol{\varphi}}$, and $\mathbf{V}_0$ are the prescribed vector fields in $\partial\mathcal{B}_\sigma\times\mathbb{I}$, $\partial\mathcal{B}_\varphi\times\mathbb{I}$, and $\mathcal{B}$, respectively. From Eq. $(3.8)_1$, the ***Bubnov-Galerkin weighted residual form*** [16–20] can be established as

$$\int_{\mathcal{B}} [\mathrm{DIV}\,\mathbf{P}(\mathbf{X},t) + \rho_0 \mathbf{B}(\mathbf{X},t) - \rho_0 \ddot{\mathbf{u}}(\mathbf{X},t)]\cdot\delta\mathbf{u}(\mathbf{X},t)\,dV = 0 \tag{3.9}$$

where $\mathbf{u}(\mathbf{X},t) = \boldsymbol{\varphi}(\mathbf{X},t) - \mathbf{X}$ denotes the displacement vector field associated with $\boldsymbol{\varphi}(\mathbf{X},t)$, and $\delta\mathbf{u}(\mathbf{X},t)$ denotes the virtual displacement vector field. Note that the material velocity and acceleration vector fields can be written as $\dot{\boldsymbol{\varphi}}(\mathbf{X},t) = \dot{\mathbf{u}}(\mathbf{X},t) \in T_{\boldsymbol{\varphi}}\mathcal{C}$ and $\ddot{\boldsymbol{\varphi}}(\mathbf{X},t) = \ddot{\mathbf{u}}(\mathbf{X},t)$, respectively. In $\mathcal{B}\times\mathbb{I}$, the variation of the deformation gradient tensor field is given as

$$\delta\mathbf{F} = \mathrm{GRAD}\,\delta\mathbf{u} = \mathrm{GRAD}\,\delta\boldsymbol{\varphi} \tag{3.10}$$

due to $\delta\mathbf{X} = \mathbf{0}$. Note that the variation of the prescribed motion map in $\partial\mathcal{B}_\varphi \times \mathbb{I}$ vanishes, i.e., $\delta\bar{\boldsymbol{\varphi}} = \mathbf{0}$. By invoking the product rule, we have

$$\mathrm{DIV}\,\mathbf{P}\cdot\delta\mathbf{u} = \mathrm{DIV}\,(\delta\mathbf{u}\cdot\mathbf{P}) - \mathbf{P} : \mathrm{GRAD}\,\delta\mathbf{u} = \mathrm{DIV}\,(\delta\mathbf{u}\cdot\mathbf{P}) - \mathbf{P} : \delta\mathbf{F} \tag{3.11}$$

Upon substitution of Eq. (3.11) into Eq. (3.9), one can obtain the following

$$\int_{\mathcal{B}} \rho_0\ddot{\mathbf{u}}\cdot\delta\mathbf{u}\,dV + \int_{\mathcal{B}} \mathbf{P} : \delta\mathbf{F}\,dV = \int_{\mathcal{B}} \rho_0\mathbf{B}\cdot\delta\mathbf{u}\,dV + \int_{\mathcal{B}} \mathrm{DIV}\,(\delta\mathbf{u}\cdot\mathbf{P})\,dV \tag{3.12}$$

Recalling the Gauss theorem, Eq. (3.12) yields the ***principle of virtual work*** for elastodynamical systems as

$$\int_{\mathcal{B}} \rho_0\ddot{\mathbf{u}}\cdot\delta\mathbf{u}\,dV + \int_{\mathcal{B}} \mathbf{P} : \delta\mathbf{F}\,dV = \int_{\mathcal{B}} \rho_0\mathbf{B}\cdot\delta\mathbf{u}\,dV + \int_{\partial\mathcal{B}_\sigma} \bar{\mathbf{T}}\cdot\delta\mathbf{u}\,dA \tag{3.13}$$

where we have used the natural and essential boundary conditions given in Eq. $(3.8)_2$ and Eq. $(3.8)_3$. That is, Eq. (3.13) implicitly contains both natural and essential boundary conditions as well as the governing equation. Of course, Eq. (3.13) can be also written as

$$\int_{\mathcal{B}} \rho_0\ddot{\boldsymbol{\varphi}}\cdot\delta\boldsymbol{\varphi}\,dV + \int_{\mathcal{B}} \mathbf{P} : \delta\mathbf{F}\,dV = \int_{\mathcal{B}} \rho_0\mathbf{B}\cdot\delta\boldsymbol{\varphi}\,dV + \int_{\partial\mathcal{B}_\sigma} \bar{\mathbf{T}}\cdot\delta\boldsymbol{\varphi}\,dA \tag{3.14}$$

Eq. (3.13) or (3.14) is weaker than Eq. (3.9) in the sense of continuity requirements for the displacement-based finite element method. Therefore, the weak form for elastodynamical systems may consist of Eq. (3.13) and the initial conditions as follows:

$$\boxed{\begin{aligned} &\int_{\mathcal{B}} \rho_0\ddot{\boldsymbol{\varphi}}\cdot\delta\boldsymbol{\varphi}\,dV + \int_{\mathcal{B}} \mathbf{P} : \delta\mathbf{F}\,dV = \int_{\mathcal{B}} \rho_0\mathbf{B}\cdot\delta\boldsymbol{\varphi}\,dV + \int_{\partial\mathcal{B}_\sigma} \bar{\mathbf{T}}\cdot\delta\boldsymbol{\varphi}\,dA \\ &\qquad\qquad \boldsymbol{\varphi}(\mathbf{X}, t_0) = \boldsymbol{\varphi}_0 = \mathbf{X} \qquad\qquad \text{in } \mathcal{B} \\ &\qquad\qquad \dot{\boldsymbol{\varphi}}(\mathbf{X}, t_0) = \dot{\boldsymbol{\varphi}}_0 = \mathbf{V}_0 \qquad\qquad \text{in } \mathcal{B} \end{aligned}} \tag{3.15}$$

3.1.2 Spatial Discretization by the Finite Element Method

Following the conventional Galerkin finite element method [21, 22], consider the admissible trial function represented by

$$\tilde{\boldsymbol{\varphi}}\left(\mathbf{X},t\right)\simeq\tilde{\boldsymbol{\varphi}}^{h}\left(\mathbf{X},t\right)=\mathbf{X}+\sum_{i=1}^{n_{\text{node}}}N_{i}\left(\mathbf{X}\right)\mathbf{q}^{i}\left(t\right)\in\mathcal{C}^{h} \tag{3.16}$$

where n_{node} denotes the number of nodes, $N_i(\mathbf{X}):\mathcal{B}\rightarrow\mathbb{R}$ denotes the prescribed shape functions which satisfy the completeness condition $\sum_{i=1}^{n_{\text{node}}}N_i\left(\mathbf{X}\right)=1$, and $\mathbf{q}^i\left(t\right)$ denote the nodal displacement vector associated with the i^{th} node at time $t\in\mathbb{I}$. The trial function $\tilde{\boldsymbol{\varphi}}^h$ is of class $C^{m-1}\left(\mathcal{B}\right)$ over the entire domain, and it belongs to the subspace

$$\mathcal{C}^{h}=\left\{\begin{array}{l}\tilde{\boldsymbol{\varphi}}^{h}\left(\mathbf{X},t\right):\mathcal{B}\times\mathbb{I}\rightarrow\mathbb{R}^{3}|\tilde{\boldsymbol{\varphi}}^{h}\in W_{2}^{m}\left(\mathcal{B}\right)\subset C^{m-1}\left(\mathcal{B}\right),\text{ in }\mathcal{B}\\ \tilde{\boldsymbol{\varphi}}^{h}=\bar{\boldsymbol{\varphi}},\text{ on }\partial\mathcal{B}_{\varphi}\end{array}\right\}\subset\mathcal{C} \tag{3.17}$$

Here, the configuration manifold for the nodal displacement vector may be defined as

$$Q:=\{\mathbf{q}(t)=(q^{1}(t),q^{2}(t),\cdots,q^{n_{\text{node}}}(t)):\mathbb{I}\rightarrow\mathbb{R}^{n_{\text{dof}}}\} \tag{3.18}$$

where $n_{\text{dof}}=3n_{\text{node}}$ denotes the number of degrees of freedom. Note that the variation of the admissible trial function $\tilde{\boldsymbol{\varphi}}^h\left(\mathbf{X},t\right):\mathcal{B}\times\mathbb{I}\rightarrow\mathcal{C}^h$ yields

$$\delta\tilde{\boldsymbol{\varphi}}^{h}\left(\mathbf{X},t\right)=\sum_{i=1}^{n_{\text{node}}}N_{i}\left(\mathbf{X}\right)\delta\mathbf{q}^{i}\left(t\right) \tag{3.19}$$

and the material velocity and acceleration vector fields are approximated as follows:

$$\begin{aligned}\tilde{\mathbf{V}}\left(\mathbf{X},t\right)\equiv\dot{\tilde{\boldsymbol{\varphi}}}\left(\mathbf{X},t\right)\simeq\dot{\tilde{\boldsymbol{\varphi}}}^{h}\left(\mathbf{X},t\right)=\sum_{i=1}^{n_{\text{node}}}\mathbf{N}_{i}\left(\mathbf{X}\right)\dot{\mathbf{q}}^{i}\left(t\right)\in T_{\boldsymbol{\varphi}}\mathcal{C}^{h}\\ \tilde{\mathbf{A}}\left(\mathbf{X},t\right)\equiv\ddot{\tilde{\boldsymbol{\varphi}}}\left(\mathbf{X},t\right)\simeq\ddot{\tilde{\boldsymbol{\varphi}}}^{h}\left(\mathbf{X},t\right)=\sum_{i=1}^{n_{\text{node}}}\mathbf{N}_{i}\left(\mathbf{X}\right)\ddot{\mathbf{q}}^{i}\left(t\right)\end{aligned} \tag{3.20}$$

where $\dot{\mathbf{q}}^i\left(t\right)\in T_{\mathbf{q}}Q$ and $\ddot{\mathbf{q}}^i\left(t\right)$ denote the nodal velocity and acceleration vectors associated with the i^{th} node at time $t\in\mathbb{I}$, respectively. Substituting Eq. (3.16)

and Eq. (3.20) into Eq. (3.13) yields

$$\sum_{i=1}^{n_{\text{node}}} \left[\sum_{j=1}^{n_{\text{node}}} M_{ij} \ddot{\mathbf{q}}^j + \mathbf{F}_i^{\text{int}} - \mathbf{F}_i^{\text{ext}} \right] \cdot \delta \mathbf{q}^i = 0 \tag{3.21}$$

where the global symmetric mass matrix is defined as $M_{ij} = \int_{\mathcal{B}} \rho_0 N_i N_j \, dV$, and the internal and external force vectors associated with the i^{th} node at time $t \in \mathbb{I}$ are given as

$$\mathbf{F}_i^{\text{int}} = \int_{\mathcal{B}} \mathbf{P}^h \text{GRAD } N_i \, dV \tag{3.22}$$

$$\mathbf{F}_i^{\text{ext}} = \int_{\mathcal{B}} \rho_0 N_i \mathbf{B} \, dV + \int_{\partial \mathcal{B}_\sigma} N_i \bar{\mathbf{T}} \, dA \tag{3.23}$$

According to Nanson's formula, the first Piola-Kirchhoff stress tensor field in space-discrete systems may be expressed as $\mathbf{P}^h = \mathbf{F}^h \cdot \mathbf{S}^h$ where $\mathbf{S}^h(\mathbf{X}, t)$ denotes the space-discrete second Piola-Kirchhoff stress second-order tensor field. Since the space-discrete deformation gradient tensor field can be written as

$$\mathbf{F}^h(\mathbf{X}, t) = \text{GRAD } \boldsymbol{\varphi}^h(\mathbf{X}, t) = \mathbf{I} + \sum_{i=1}^{n_{\text{node}}} \mathbf{q}^i(t) \otimes \text{GRAD } N_i(\mathbf{X}) \tag{3.24}$$

$\mathbf{I}$ denotes the identity second-order tensor, the space-discrete symmetric second Piola-Kirchhoff stress tensor field is given as

$$\mathbf{S}^h(\mathbf{X}, t) = \bar{\mathbf{S}}^h(\mathbf{F}^h) \tag{3.25}$$

where $\bar{\mathbf{S}}^h(\mathbf{F}^h)$ is the stress response function for $\mathbf{S}(\mathbf{X}, t)$ in terms of $\mathbf{F}^h$. Hence, Eq. (3.22) can be also written as

$$\mathbf{F}_i^{\text{int}} = \int_{\mathcal{B}} \mathbf{F}^h \cdot \bar{\mathbf{S}}^h(\mathbf{F}^h) \text{ GRAD } N_i \, dV \tag{3.26}$$

From Eq. (3.26), we can also show the internal force vector as

$$\mathbf{F}_i^{\text{int}} = \sum_{j=1}^{n_{\text{node}}} \int_{\mathcal{B}} \text{GRAD } N_i \cdot \bar{\mathbf{S}}^h(\mathbf{F}^h) \text{GRAD } N_j dV \mathbf{q}^j \tag{3.27}$$

Since the set $\delta\boldsymbol{q} = (\delta q^1, \delta q^2, \cdots, \delta q^{n_{\mathrm{dof}}})^T$ is linearly independent, the terms enclosed in the parenthesis in Eq. (3.21) can be regarded as individual coefficients [6,23–25]. Therefore, Eq. (3.21) stands for a linear combination of functions in the sense of functional analysis [26,27]. Consequently, since $\delta\boldsymbol{q}$ is arbitrary, we obtain

$$\sum_{j=1}^{n_{\mathrm{node}}} M_{ij}\ddot{\mathbf{q}}^j + \mathbf{F}_i^{\mathrm{int}} - \mathbf{F}_i^{\mathrm{ext}} = \mathbf{0} \tag{3.28}$$

Eq. (3.28) is the semi-discrete second-order system of ordinary differential equations. The initial-value problem for semi-discrete elastodynamical systems consists of Eq. (3.28) and the initial conditions:

$$\boxed{\begin{aligned} \sum_{j=1}^{n_{\mathrm{node}}} M_{ij}\ddot{\mathbf{q}}^j + \mathbf{F}_i^{\mathrm{int}} - \mathbf{F}_i^{\mathrm{ext}} &= \mathbf{0} \\ \mathbf{q}(t_0) &= \mathbf{q}_0 \\ \dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0 \end{aligned}} \tag{3.29}$$

where $\mathbf{q}_0 = (q_0^1, q_0^2, \cdots, q_0^{n_{\mathrm{dof}}})^T$ and $\dot{\mathbf{q}}_0 = (\dot{q}_0^1, \dot{q}_0^2, \cdots, \dot{q}_0^{n_{\mathrm{dof}}})^T$ are the given nodal displacement and velocity vectors at the initial time t_0. These are finally integrated in time to find the dynamic response.

3.2 Scalar Formalisms: Strong Forms of the Initial Boundary-Value Problems and Weak Forms for Elastodynamical Systems

The derivation of the strong and weak forms of Cauchy's first equation of motion in the sense of the scalar formalisms, namely, from Hamilton's principle and from the principle of balance of total energy for an isothermal continuum body are briefly summarized. In the scalar-formalisms, unlike the vector formalism, we mainly deal with the scalar-valued functions; that is, the autonomous Lagrangian function and the autonomous total energy function of an isothermal continuum

body, which is sometimes called the autonomous mechanical energy in the literature; and the autonomous Hamiltonian function, which is equivalent to the autonomous mechanical energy function. It is very important to note that both Hamilton's principle and the principle of balance of mechanical energy are consequences of Cauchy's first equation of motion. Once the Cauchy's first equation of motion in the Lagrangian and Hamiltonian mechanics are derived, the spatially-discrete forms are readily obtained following the Galerkin finite element method.

3.2.1 Hamilton's Principle and Modified Hamilton's Principle - Variational Calculus Setting

Hamilton's Principle: Lagrangian Mechanics

The tangent bundle (infinite dimensional state space) associated with $\mathcal{C}$, which is described in Equation 3.1, may be defined as

$$T\mathcal{C} := \left\{ (\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}) : \mathcal{B} \times \mathbb{I} \to \mathbb{R}^3 \times \mathbb{R}^3 | \boldsymbol{\varphi} \in \mathcal{C}, \ \ \dot{\boldsymbol{\varphi}} \in T_{\boldsymbol{\varphi}}\mathcal{C}, \text{ and } \ \dot{\boldsymbol{\varphi}}|_{\partial\mathcal{B}_\varphi} = \mathbf{0} \right\} \tag{3.30}$$

Define an autonomous Lagrangian $\mathscr{L}(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}) : T\mathcal{C} \to \mathbb{R}$ with dead loads as

$$\mathscr{L}(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}) := \int_{\mathcal{B}} \mathcal{L}(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}, \mathbf{F}) \, dV + \int_{\partial\mathcal{B}_\sigma} \bar{\mathbf{T}}(\mathbf{X}, t, \hat{\mathbf{N}}) \cdot \boldsymbol{\varphi}(\mathbf{X}, t) \, dA \tag{3.31}$$

where $\bar{\mathbf{T}}(\mathbf{X}, t, \hat{\mathbf{N}})$ denotes the prescribed material traction vector field on $\partial\mathcal{B}_\sigma$ at time $t \in \mathbb{I}$, $\hat{\mathbf{N}}(\mathbf{X}, t)$ denotes the outward unit normal vector field in the material description on $\partial\mathcal{B}_\sigma$ at $t \in \mathbb{I}$, and $\mathcal{L}(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}, \mathbf{F})$ denotes the Lagrangian density function given by

$$\mathcal{L}(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}, \mathbf{F}) := \frac{1}{2}\rho_0 \dot{\boldsymbol{\varphi}}(\mathbf{X}, t) \cdot \dot{\boldsymbol{\varphi}}(\mathbf{X}, t) - \rho_0 \bar{\Psi}(\mathbf{F}) + \rho_0 \mathbf{B} \cdot \boldsymbol{\varphi}(\mathbf{X}, t) \tag{3.32}$$

In the above, $\bar{\Psi}(\mathbf{F})$ denotes the response function in terms of $\mathbf{F}$ for the Helmholtz free energy per unit mass in homogeneous materials as an elastic potential function Ψ in $\mathcal{B}$, $\rho_0(\mathbf{X}) \equiv \rho(\mathbf{X}, t_0)$ denotes the material densify, and $\mathbf{B}(\mathbf{X}, t)$ denotes the material body force vector field. Define the action map $S : \Xi(\mathcal{C}) \to \mathbb{R}$ as

$$S[\boldsymbol{\varphi}] := \int_{t_0}^{t_N} \mathscr{L}(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}) \, dt \tag{3.33}$$

defined on the smooth manifold

$$\Xi(\mathcal{C}) = \Xi(\mathbb{I}, \mathcal{C}) = \left\{ \boldsymbol{\varphi} : \mathcal{B} \times \mathbb{I} \to \mathcal{C} \ \mid \boldsymbol{\varphi} \text{ is a } C^2 \text{ curve.} \right\} \tag{3.34}$$

Hamilton's principle reads as

$$\boxed{\begin{aligned} \delta S &= \delta \int_{t_0}^{t_N} \mathscr{L}\left(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}\right) dt \\ &= \delta \int_{t_0}^{t_N} \int_{\mathcal{B}} \mathcal{L}\left(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}, \mathbf{F}\right) dV dt + \delta \int_{t_0}^{t_N} \int_{\partial \mathcal{B}_\sigma} \bar{\mathbf{T}} \cdot \boldsymbol{\varphi}\left(\mathbf{X}, t\right) dA dt = 0 \\ \text{with } & \ \delta\boldsymbol{\varphi}(\mathbf{X}, t_0) = \delta\boldsymbol{\varphi}(\mathbf{X}, t_N) = \mathbf{0} \end{aligned}} \tag{3.35}$$

for holonomic systems with dead loads. The boundary conditions, $\delta\boldsymbol{\varphi}(\mathbf{X}, t_0) = \delta\boldsymbol{\varphi}(\mathbf{X}, t_N) = \mathbf{0}$, imply that the configurations at both t_0 and t_N must be prescribed. Hence, Eq. (3.35) yields

$$\begin{aligned} \delta S &= \int_{t_0}^{t_N} \int_{\mathcal{B}} \delta\mathcal{L}\left(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}, \mathbf{F}\right) dV dt + \int_{t_0}^{t_N} \int_{\partial \mathcal{B}_\sigma} \bar{\mathbf{T}} \cdot \delta\boldsymbol{\varphi}\left(\mathbf{X}, t\right) dA dt \\ &= \int_{t_0}^{t_N} \int_{\mathcal{B}} \left[\frac{\partial \mathcal{L}}{\partial \boldsymbol{\varphi}} - \frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\varphi}}} \right) - \left(\mathrm{DIV}\, \frac{\partial \mathcal{L}}{\partial \mathbf{F}} \right) \right] \cdot \delta\boldsymbol{\varphi} dV dt \\ &+ \int_{t_0}^{t_N} \int_{\mathcal{B}} \mathrm{DIV}\, \left(\delta\boldsymbol{\varphi} \cdot \frac{\partial \mathcal{L}}{\partial \mathbf{F}} \right) dV dt \\ &+ \int_{t_0}^{t_N} \int_{\partial \mathcal{B}_\sigma} \bar{\mathbf{T}} \cdot \delta\boldsymbol{\varphi}\, dA dt + \int_{\mathcal{B}} \frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\varphi}}} \cdot \delta\boldsymbol{\varphi} dV \Big|_{t_0}^{t_N} = 0 \end{aligned} \tag{3.36}$$

Using the Gauss theorem and the boundary conditions, $\delta\boldsymbol{\varphi}(\mathbf{X}, t_0) = \delta\boldsymbol{\varphi}(\mathbf{X}, t_N) = \mathbf{0}$,

$$\begin{aligned} \delta S = &\int_{t_0}^{t_N} \int_{\mathcal{B}} \left[\frac{\partial \mathcal{L}}{\partial \boldsymbol{\varphi}} - \left(\mathrm{DIV}\, \frac{\partial \mathcal{L}}{\partial \mathbf{F}} \right) - \frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\varphi}}} \right) \right] \cdot \delta\boldsymbol{\varphi}\, dV dt \\ &+ \int_{t_0}^{t_N} \int_{\partial \mathcal{B}_\sigma} \left[\bar{\mathbf{T}} + \frac{\partial \mathcal{L}}{\partial \mathbf{F}} \cdot \hat{\mathbf{N}} \right] \cdot \delta\boldsymbol{\varphi}\, dA dt = 0 \end{aligned} \tag{3.37}$$

Since $\delta\boldsymbol{\varphi} \in T_{\boldsymbol{\varphi}}\Xi(\mathcal{C})$ is arbitrary, we obtain

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\varphi}}} \right) - \frac{\partial \mathcal{L}}{\partial \boldsymbol{\varphi}} = \mathrm{DIV}\, \frac{\partial \mathcal{L}}{\partial \mathbf{F}} \quad \text{in } \ \mathcal{B} \times \mathbb{I} \tag{3.38}$$

$$\bar{\mathbf{T}} = -\frac{\partial \mathcal{L}}{\partial \mathbf{F}} \cdot \hat{\mathbf{N}} \quad \text{in } \ \partial\mathcal{B}_\sigma \times \mathbb{I} \tag{3.39}$$

Note that, upon substituting the Lagrangian density function defined in Eq. (3.32), the right-hand side of Eq. (3.39) can be written as

$$-\frac{\partial \mathcal{L}}{\partial \mathbf{F}} \cdot \hat{\mathbf{N}} = \rho_0 \frac{\partial \bar{\Psi}\left(\mathbf{F}\right)}{\partial \mathbf{F}} \cdot \hat{\mathbf{N}} = \frac{\partial \bar{W}\left(\mathbf{F}\right)}{\partial \mathbf{F}} \cdot \hat{\mathbf{N}} = \bar{\mathbf{P}}(\mathbf{F}) \cdot \hat{\mathbf{N}} \tag{3.40}$$

where $\bar{W}(\mathbf{F}) \in \mathbb{R}$ and $\bar{\mathbf{P}}(\mathbf{F})$ are the response functions in terms of $\mathbf{F}$ for the strain energy W and the first Piola-Kirchhoff stress second-order tensor field $\mathbf{P}(\mathbf{X}, t)$, respectively. By substituting the Lagrangian density function defined in Eq. (3.32) into Eq. (3.38), Cauchy's first law of motion [10, 11] or simply Cauchy's equation of motion [12–15] is obtained as

$$\rho_0 \ddot{\boldsymbol{\varphi}}\left(\mathbf{X}, t\right) = \mathrm{DIV}\, \mathbf{P}\left(\mathbf{X}, t\right) + \rho_0 \mathbf{B}\left(\mathbf{X}, t\right) \tag{3.41}$$

in $\mathcal{B} \times \mathbb{I}$. Therefore, the strong form of the initial boundary-value problem for elastodynamical systems has been obtained from Hamilton's principle as follows: Find the motion map $\boldsymbol{\varphi}(\mathbf{X}, t) : \mathcal{B} \times \mathbb{I} \to \mathcal{S}$, which is the dependent variable, from

$$\boxed{\begin{aligned} \frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\varphi}}}\right) - \frac{\partial \mathcal{L}}{\partial \boldsymbol{\varphi}} &= \mathrm{DIV}\, \frac{\partial \mathcal{L}}{\partial \mathbf{F}} && \text{in } \mathcal{B} \times \mathbb{I} \\ \bar{\mathbf{T}}(\mathbf{X}, t, \hat{\mathbf{N}}) &= \mathbf{P}\left(\mathbf{X}, t\right) \cdot \hat{\mathbf{N}}\left(\mathbf{X}, t\right) && \text{in } \partial\mathcal{B}_\sigma \times \mathbb{I} \\ \boldsymbol{\varphi} &= \bar{\boldsymbol{\varphi}} && \text{in } \partial\mathcal{B}_\varphi \times \mathbb{I} \\ \boldsymbol{\varphi}\left(\mathbf{X}, t_0\right) &= \boldsymbol{\varphi}_0 = \mathbf{X} && \text{in } \mathcal{B} \\ \dot{\boldsymbol{\varphi}}\left(\mathbf{X}, t_0\right) &= \dot{\boldsymbol{\varphi}}_0 = \mathbf{V}_0 && \text{in } \mathcal{B} \end{aligned}} \tag{3.42}$$

where Eq. $(3.42)_4$ and Eq. $(3.42)_5$ are the initial conditions at time t_0.

Weak Form Note that we can obtain the following expression from Eq. (3.37),

$$\begin{aligned} &\int_{t_0}^{t_N} \int_{\mathcal{B}} \frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\varphi}}}\right) \cdot \delta\boldsymbol{\varphi}\, dV dt - \int_{t_0}^{t_N} \int_{\mathcal{B}} \frac{\partial \mathcal{L}}{\partial \mathbf{F}} : \delta\mathbf{F}\, dV dt \\ &- \int_{t_0}^{t_N} \int_{\mathcal{B}} \frac{\partial \mathcal{L}}{\partial \boldsymbol{\varphi}} \cdot \delta\boldsymbol{\varphi}\, dV dt - \int_{t_0}^{t_N} \int_{\partial\mathcal{B}_\sigma} \bar{\mathbf{T}} \cdot \delta\boldsymbol{\varphi}\, dA dt = 0 \end{aligned} \tag{3.43}$$

Here, the variation of the deformation gradient tensor field is given as $\delta\mathbf{F} = \mathrm{GRAD}\, \delta\boldsymbol{\varphi}$. Assuming the integrand of the time integral in Eq. (3.43) is continuous

in time, we then obtain the ***Lagrangian version of the weak form*** as follows due to the localization theorem in the time domain:

$$\boxed{\begin{aligned} \int_{\mathcal{B}} \frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\varphi}}}\right) \cdot \delta\boldsymbol{\varphi}\, dV - \int_{\mathcal{B}} \frac{\partial \mathcal{L}}{\partial \mathbf{F}} : \delta\mathbf{F}\, dV - \int_{\mathcal{B}} \frac{\partial \mathcal{L}}{\partial \boldsymbol{\varphi}} \cdot \delta\boldsymbol{\varphi}\, dV - \int_{\partial\mathcal{B}_\sigma} \bar{\mathbf{T}} \cdot \delta\boldsymbol{\varphi}\, dA = 0 \\ \boldsymbol{\varphi}\left(\mathbf{X}, t_0\right) = \boldsymbol{\varphi}_0 = \mathbf{X} \quad \text{in } \mathcal{B} \\ \dot{\boldsymbol{\varphi}}\left(\mathbf{X}, t_0\right) = \dot{\boldsymbol{\varphi}}_0 = \mathbf{V}_0 \quad \text{in } \mathcal{B} \end{aligned}} \tag{3.44}$$

Note that Eq. $(3.44)_1$ yields the principle of virtual work in elastodynamical systems for dead load cases. Eq. (3.44) can be used to derive the semi-discrete equations of motion via the finite element method as shown in a later section.

Remark 2

1. ***Representations with Autonomous Mechanical Energy:*** *An autonomous total energy, $\mathscr{E}(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}) : T\mathcal{C} \to \mathbb{R}$, for isothermal elastodynamical systems, i.e., autonomous mechanical energy, may be defined as*

$$\mathscr{E}\left(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}\right) := \int_{\mathcal{B}} \mathcal{E}\left(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}, \mathbf{F}\right) dV - \int_{\partial\mathcal{B}_\sigma} \bar{\mathbf{T}}(\mathbf{X}, t, \hat{\mathbf{N}}) \cdot \boldsymbol{\varphi}\left(\mathbf{X}, t\right) dA \tag{3.45}$$

where the mechanical energy density function is given by

$$\mathcal{E}\left(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}, \mathbf{F}\right) := \frac{1}{2}\rho_0 \dot{\boldsymbol{\varphi}}\left(\mathbf{X}, t\right) \cdot \dot{\boldsymbol{\varphi}}\left(\mathbf{X}, t\right) + \rho_0 \bar{\Psi}\left(\mathbf{F}\right) - \rho_0 \mathbf{B} \cdot \boldsymbol{\varphi}\left(\mathbf{X}, t\right) \tag{3.46}$$

In terms of the autonomous mechanical energy, Hamilton's principle can be represented as

$$S[\boldsymbol{\varphi}] := \int_{t_0}^{t_N} \left[\int_{\mathcal{B}} \rho_0 \dot{\boldsymbol{\varphi}} \cdot \dot{\boldsymbol{\varphi}}\, dV - \mathscr{E}\left(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}\right)\right] dt \tag{3.47}$$

and it leads to the strong form of the initial boundary value problem which

is equivalent to the system given in Eq. (3.42)*. That is,*

$$\boxed{\begin{aligned} \frac{d}{dt}\left(\frac{\partial \mathcal{E}}{\partial \dot{\boldsymbol{\varphi}}}\right) + \frac{\partial \mathcal{E}}{\partial \boldsymbol{\varphi}} &= \mathrm{DIV}\,\frac{\partial \mathcal{E}}{\partial \mathbf{F}} && \text{in } \mathcal{B}\times\mathbb{I} \\ \bar{\mathbf{T}}(\mathbf{X},t,\hat{\mathbf{N}}) &= \mathbf{P}\left(\mathbf{X},t\right)\cdot\hat{\mathbf{N}}\left(\mathbf{X},t\right) && \text{in } \partial\mathcal{B}_\sigma\times\mathbb{I} \\ \boldsymbol{\varphi} &= \bar{\boldsymbol{\varphi}} && \text{in } \partial\mathcal{B}_\varphi\times\mathbb{I} \\ \boldsymbol{\varphi}\left(\mathbf{X},t_0\right) &= \boldsymbol{\varphi}_0 = \mathbf{X} && \text{in } \mathcal{B} \\ \dot{\boldsymbol{\varphi}}\left(\mathbf{X},t_0\right) &= \dot{\boldsymbol{\varphi}}_0 = \mathbf{V}_0 && \text{in } \mathcal{B} \end{aligned}} \tag{3.48}$$

The autonomous mechanical energy version of the weak form yields

$$\boxed{\begin{aligned} &\int_{\mathcal{B}} \frac{d}{dt}\left(\frac{\partial \mathcal{E}}{\partial \dot{\boldsymbol{\varphi}}}\right)\cdot\delta\boldsymbol{\varphi}\,dV + \int_{\mathcal{B}}\frac{\partial \mathcal{E}}{\partial \mathbf{F}} : \delta\mathbf{F}\,dV \\ &+\int_{\mathcal{B}}\frac{\partial \mathcal{E}}{\partial \boldsymbol{\varphi}}\cdot\delta\boldsymbol{\varphi}\,dV + \int_{\partial\mathcal{B}_\sigma}\bar{\mathbf{T}}\cdot\delta\boldsymbol{\varphi}\,dA = 0 \\ &\boldsymbol{\varphi}\left(\mathbf{X},t_0\right) = \boldsymbol{\varphi}_0 = \mathbf{X} \quad \text{in } \mathcal{B} \\ &\dot{\boldsymbol{\varphi}}\left(\mathbf{X},t_0\right) = \dot{\boldsymbol{\varphi}}_0 = \mathbf{V}_0 \quad \text{in } \mathcal{B} \end{aligned}} \tag{3.49}$$

Modified Hamilton's Principle: Hamiltonian Mechanics

The autonomous Hamiltonian for dead loads may be defined in the cotangent bundle (phase space), $T_{\boldsymbol{\varphi}}\mathcal{C}$, associated with the configuration manifold $\mathcal{C}$ as

$$\mathscr{H}\left(\boldsymbol{\phi},\boldsymbol{\varphi}\right) = \int_{\mathcal{B}}\mathcal{H}(\boldsymbol{\phi},\boldsymbol{\varphi},\mathbf{F})dV - \int_{\partial\mathcal{B}_\sigma}\bar{\mathbf{T}}(\mathbf{X},t,\hat{\mathbf{N}})\cdot\boldsymbol{\varphi}\left(\mathbf{X},t\right)dA \tag{3.50}$$

where the Hamiltonian density function is obtained via the Legendre transformation as

$$\begin{aligned} \mathcal{H}\left(\boldsymbol{\phi},\boldsymbol{\varphi},\mathbf{F}\right) &= \boldsymbol{\phi}\cdot\dot{\boldsymbol{\varphi}} - \mathcal{L}\left(\boldsymbol{\varphi},\dot{\boldsymbol{\varphi}},\mathbf{F}\right) \\ &= \frac{1}{2\rho_0}\boldsymbol{\phi}\left(\mathbf{X},t\right)\cdot\boldsymbol{\phi}\left(\mathbf{X},t\right) + \rho_0\bar{\Psi}\left(\mathbf{F}\right) - \rho_0\mathbf{B}\left(\mathbf{X},t\right)\cdot\boldsymbol{\varphi}\left(\mathbf{X},t\right) \end{aligned} \tag{3.51}$$

where $\boldsymbol{\phi}\left(\mathbf{X},t\right) = \rho_0\dot{\boldsymbol{\varphi}}\left(\mathbf{X},t\right) : \mathcal{B}\times\mathbb{I} \to T^*_{\boldsymbol{\varphi}}\mathcal{C}$ denotes the canonical momentum defined in the cotangent space, which is dual to the tangent space $T_{\boldsymbol{\varphi}}\mathcal{C}$,

$$T^*_{\boldsymbol{\varphi}}\mathcal{C} := \left\{\boldsymbol{\phi} : \mathcal{B}\times\mathbb{I}\to\mathbb{R}^3\right\} \tag{3.52}$$

The resulting action map in Hamiltonian mechanics may be defined as

$$S[\boldsymbol{\varphi}, \boldsymbol{\phi}] := \int_{t_0}^{t_N} \left[\int_{\mathcal{B}} \boldsymbol{\phi} \cdot \dot{\boldsymbol{\varphi}} \, dV - \mathscr{H}\left(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}\right) \right] dt \tag{3.53}$$

in terms of the primary variables $\boldsymbol{\varphi} \in \mathcal{C}$ and $\boldsymbol{\phi}$. In Hamiltonian mechanics, the ***modified Hamilton's principle*** may read as

$$\boxed{\begin{aligned} \delta S =& \delta \int_{t_0}^{t_N} \left[\int_{\mathcal{B}} \boldsymbol{\phi} \cdot \dot{\boldsymbol{\varphi}} \, dV - \mathscr{H}\left(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}\right) \right] dt \\ =& \delta \int_{t_0}^{t_N} \int_{\mathcal{B}} \boldsymbol{\phi} \cdot \dot{\boldsymbol{\varphi}} \, dV dt - \delta \int_{t_0}^{t_N} \int_{\mathcal{B}} \mathcal{H}\left(\boldsymbol{\varphi}, \boldsymbol{\phi}, \mathbf{F}\right) dV dt \\ &+ \delta \int_{t_0}^{t_N} \int_{\partial \mathcal{B}_\sigma} \bar{\mathbf{T}} \cdot \boldsymbol{\varphi}\left(\mathbf{X}, t\right) dA dt = 0 \\ \text{with } & \delta\boldsymbol{\varphi}(\mathbf{X}, t_0) = \delta\boldsymbol{\varphi}(\mathbf{X}, t_N) = \mathbf{0} \end{aligned}} \tag{3.54}$$

which leads to

$$\begin{aligned} \delta S = & \int_{t_0}^{t_N} \int_{\mathcal{B}} \left[\left(\dot{\boldsymbol{\varphi}} - \frac{\partial \mathcal{H}}{\partial \boldsymbol{\phi}} \right) \cdot \delta\boldsymbol{\phi} - \left(\dot{\boldsymbol{\phi}} + \frac{\partial \mathcal{H}}{\partial \boldsymbol{\varphi}} - \mathrm{DIV} \, \frac{\partial \mathcal{H}}{\partial \mathbf{F}} \right) \cdot \delta\boldsymbol{\varphi} \right] dV dt \\ & + \int_{t_0}^{t_N} \int_{\partial \mathcal{B}_\sigma} \left(\bar{\mathbf{T}} - \frac{\partial \mathcal{H}}{\partial \mathbf{F}} \cdot \hat{\mathbf{N}} \right) \cdot \delta\boldsymbol{\varphi} \, dA dt \\ & + \left[\int_{\mathcal{B}} \boldsymbol{\phi} \cdot \delta\boldsymbol{\varphi} \, dV \right] \Big|_{t_0}^{t_N} = 0 \end{aligned} \tag{3.55}$$

Due to the boundary conditions, $\delta\boldsymbol{\varphi}(\mathbf{X}, t_0) = \delta\boldsymbol{\varphi}(\mathbf{X}, t_N) = \mathbf{0}$, the last term in Eq. (3.55) vanishes. Since $\delta\boldsymbol{\varphi}$ and $\delta\boldsymbol{\phi}$ are arbitrary, we readily obtain

$$\dot{\boldsymbol{\phi}} = -\frac{\partial \mathcal{H}}{\partial \boldsymbol{\varphi}} + \mathrm{DIV} \, \frac{\partial \mathcal{H}}{\partial \mathbf{F}} \tag{3.56}$$

$$\dot{\boldsymbol{\varphi}} = \frac{\partial \mathcal{H}}{\partial \boldsymbol{\phi}} \tag{3.57}$$

in $\mathcal{B} \times \mathbb{I}$, with

$$\bar{\mathbf{T}} = \frac{\partial \mathcal{H}}{\partial \mathbf{F}} \cdot \hat{\mathbf{N}} = \rho_0 \frac{\partial \bar{\Psi}\left(\mathbf{F}\right)}{\partial \mathbf{F}} \cdot \hat{\mathbf{N}} = \frac{\partial \bar{W}\left(\mathbf{F}\right)}{\partial \mathbf{F}} \cdot \hat{\mathbf{N}} = \bar{\mathbf{P}}(\mathbf{F}) \cdot \hat{\mathbf{N}} \tag{3.58}$$

in $\partial\mathcal{B}_\sigma \times \mathbb{I}$. Upon substitution of the Hamilton density function given in Eq. (3.51) into Eq. (3.56) and Eq. (3.57), the Cauchy's equations of motion in $(\boldsymbol{\varphi}, \boldsymbol{\phi})$ in the sense of the canonical equations, i.e.,

$$\dot{\boldsymbol{\phi}}(\mathbf{X}, t) = \rho_0 \mathbf{B}(\mathbf{X}, t) + \mathrm{DIV}\, \mathbf{P}(\mathbf{X}, t) \tag{3.59}$$

$$\dot{\boldsymbol{\varphi}}(\mathbf{X}, t) = \rho_0^{-1}(\mathbf{X}) \boldsymbol{\phi}(\mathbf{X}, t) \tag{3.60}$$

are readily obtained. Therefore, the strong form of the initial boundary-value problem for elastodynamical systems has been obtained from modified Hamilton's principle as follows: Find $(\boldsymbol{\varphi}, \boldsymbol{\phi})(\mathbf{X}, t) : \mathcal{B} \times \mathbb{I} \to T^*\mathcal{C}$ from

$$\boxed{\begin{aligned}
\dot{\boldsymbol{\phi}} &= -\frac{\partial \mathcal{H}}{\partial \boldsymbol{\varphi}} + \mathrm{DIV}\, \frac{\partial \mathcal{H}}{\partial \mathbf{F}} && \text{in } \mathcal{B} \times \mathbb{I} \\
\dot{\boldsymbol{\varphi}} &= \frac{\partial \mathcal{H}}{\partial \boldsymbol{\phi}} && \text{in } \mathcal{B} \times \mathbb{I} \\
\bar{\mathbf{T}}(\mathbf{X}, t, \hat{\mathbf{N}}) &= \mathbf{P}\,(\mathbf{X}, t) \cdot \hat{\mathbf{N}}\,(\mathbf{X}, t) && \text{in } \partial\mathcal{B}_\sigma \times \mathbb{I} \\
\boldsymbol{\varphi} &= \bar{\boldsymbol{\varphi}} && \text{in } \partial\mathcal{B}_\varphi \times \mathbb{I} \\
\boldsymbol{\varphi}\,(\mathbf{X}, t_0) &= \boldsymbol{\varphi}_0 = \mathbf{X} && \text{in } \mathcal{B} \\
\boldsymbol{\phi}\,(\mathbf{X}, t_0) &= \boldsymbol{\phi}_0 && \text{in } \mathcal{B}
\end{aligned}} \tag{3.61}$$

Weak Form From Eq. (3.55), we get

$$\begin{aligned}
&\int_{t_0}^{t_N} \int_{\mathcal{B}} \dot{\boldsymbol{\phi}} \cdot \delta\boldsymbol{\varphi}\, dV dt + \int_{t_0}^{t_N} \int_{\mathcal{B}} \frac{\partial \mathcal{H}}{\partial \mathbf{F}} : \delta\mathbf{F}\, dV dt \\
&+ \int_{t_0}^{t_N} \int_{\mathcal{B}} \frac{\partial \mathcal{H}}{\partial \boldsymbol{\varphi}} \cdot \delta\boldsymbol{\varphi}\, dV dt - \int_{t_0}^{t_N} \int_{\partial\mathcal{B}_\sigma} \bar{\mathbf{T}} \cdot \delta\boldsymbol{\varphi}\, dA dt = 0 \\
&\int_{t_0}^{t_N} \int_{\mathcal{B}} \dot{\boldsymbol{\varphi}} \cdot \delta\boldsymbol{\phi}\, dV dt - \int_{t_0}^{t_N} \int_{\partial\mathcal{B}_\sigma} \frac{\partial \mathcal{H}}{\partial \boldsymbol{\varphi}} \cdot \delta\boldsymbol{\phi}\, dA dt = 0
\end{aligned} \tag{3.62}$$

Assuming the integrands of the time integral in Eq. (3.62) are continuous in time, we then obtain the ***Hamiltonian version of the weak form*** as follows due

to the localization theorem in the time domain:

$$\boxed{\begin{aligned}\int_{\mathcal{B}} \dot{\boldsymbol{\phi}} \cdot \delta\boldsymbol{\varphi}\, dV + \int_{\mathcal{B}} \frac{\partial \mathcal{H}}{\partial \mathbf{F}} : \delta\mathbf{F}\, dV + \int_{\mathcal{B}} \frac{\partial \mathcal{H}}{\partial \boldsymbol{\varphi}} \cdot \delta\boldsymbol{\varphi}\, dV - \int_{\partial\mathcal{B}_\sigma} \bar{\mathbf{T}} \cdot \delta\boldsymbol{\varphi}\, dA = 0 \\ \int_{\mathcal{B}} \dot{\boldsymbol{\varphi}} \cdot \delta\boldsymbol{\phi}\, dV - \int_{\partial\mathcal{B}_\sigma} \frac{\partial \mathcal{H}}{\partial \boldsymbol{\varphi}} \cdot \delta\boldsymbol{\phi}\, dA = 0 \\ \boldsymbol{\varphi}(\mathbf{X}, t_0) = \boldsymbol{\varphi}_0 = \mathbf{X} \qquad \text{in } \mathcal{B} \\ \boldsymbol{\phi}(\mathbf{X}, t_0) = \boldsymbol{\phi}_0 \qquad \text{in } \mathcal{B}\end{aligned}} \tag{3.63}$$

Eq. (3.63) can be employed to derive the semi-discrete canonical equations of motion via the finite element method as shown in a later section.

3.2.2 Principle of Balance of Mechanical Energy: Differential Calculus Setting

Alternatively, we show the development of the strong and weak forms of the equation(s) of motion directly starting from the principle of balance of mechanical energy in the sense of differential calculus setting (not vector calculus setting).

Total Energy Representation of Equation of Motion and Framework

Consider the autonomous mechanical energy and the associated mechanical energy density function defined in Eq. (3.45) and Eq. (3.46), respectively. In isothermal elastodynamical conservative systems with dead loads, the ***principle of balance of mechanical energy*** (conservation of mechanical energy in this case) reads as

$$\boxed{\frac{d}{dt}\mathscr{E}(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}) = \frac{d}{dt}\int_{\mathcal{B}} \mathcal{E}(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}, \mathbf{F})\, dV - \frac{d}{dt}\int_{\partial\mathcal{B}_\sigma} \bar{\mathbf{T}} \cdot \boldsymbol{\varphi}(\mathbf{X}, t)\, dA = 0} \tag{3.64}$$

Eq. (3.64) leads to

$$\begin{aligned}\frac{d}{dt}\mathscr{E}(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}) = \int_{\mathcal{B}} \left[\frac{d}{dt}\left(\frac{\partial \mathcal{E}}{\partial \dot{\boldsymbol{\varphi}}}\right) + \frac{\partial \mathcal{E}(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}, \mathbf{F})}{\partial \boldsymbol{\varphi}} - \mathrm{DIV}\, \frac{\partial \mathcal{E}}{\partial \mathbf{F}}\right] \cdot \dot{\boldsymbol{\varphi}}(\mathbf{X}, t)\, dV \\ - \int_{\partial\mathcal{B}_\sigma} \left(\bar{\mathbf{T}} - \frac{\partial \mathcal{E}}{\partial \mathbf{F}} \cdot \hat{\mathbf{N}}\right) \cdot \dot{\boldsymbol{\varphi}}(\mathbf{X}, t)\, dA = 0\end{aligned} \tag{3.65}$$

Since the material velocity vector field $\dot{\boldsymbol{\varphi}}(\mathbf{X}, t)$ is *linearly independent* in $\mathcal{B}$ and on $\partial\mathcal{B}_\sigma$, we readily obtain the strong form of the initial boundary-value problem in terms of the mechanical energy density function as summarized in Eq. (3.48).

Weak Form Note that from Eq. (3.65), we can obtain the ***theorem of power expended*** for a conservative system as

$$\begin{aligned}\frac{d}{dt}\mathscr{E}(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}) = \int_{\mathcal{B}} \frac{d}{dt}\left(\frac{\partial \mathcal{E}}{\partial \dot{\boldsymbol{\varphi}}}\right) \cdot \dot{\boldsymbol{\varphi}}\, dV + \int_{\mathcal{B}} \frac{\partial \mathcal{E}}{\partial \mathbf{F}} : \dot{\mathbf{F}}\, dV \\ + \int_{\mathcal{B}} \frac{\partial \mathcal{E}}{\partial \boldsymbol{\varphi}} \cdot \dot{\boldsymbol{\varphi}}\, dV - \int_{\partial\mathcal{B}_\sigma} \bar{\mathbf{T}} \cdot \dot{\boldsymbol{\varphi}}\, dA = 0\end{aligned} \tag{3.66}$$

Here, the total time derivative of the deformation gradient field is given as $\dot{\mathbf{F}} = \mathrm{GRAD}\dot{\boldsymbol{\varphi}}$. Eq. (3.66) with the prescribed initial conditions forms the weak form which is readily amenable to space discretization.

Remark 3

1. *In terms of Lagrangian $\mathscr{L}(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}})$, the mechanical energy, which is often called as the Jacobi integral in the literature, can be written as*

$$\mathscr{E}(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}) = \frac{\partial \mathscr{L}}{\partial \dot{\boldsymbol{\varphi}}} \cdot \dot{\boldsymbol{\varphi}} - \mathscr{L} \tag{3.67}$$

and the strong form of the initial boundary-value problem summarized in equation (3.42) *is obtained as the consequence of the principle of balance of mechanical energy, i.e., equation* (3.64)*. Therefore, the weak form yields*

$$\int_{\mathcal{B}} \frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\varphi}}}\right) \cdot \dot{\boldsymbol{\varphi}}\, dV - \int_{\mathcal{B}} \frac{\partial \mathcal{L}}{\partial \mathbf{F}} : \dot{\mathbf{F}}\, dV - \int_{\mathcal{B}} \frac{\partial \mathcal{L}}{\partial \boldsymbol{\varphi}} \cdot \dot{\boldsymbol{\varphi}}\, dV - \int_{\partial\mathcal{B}_\sigma} \bar{\mathbf{T}} \cdot \dot{\boldsymbol{\varphi}}\, dA = 0 \tag{3.68}$$

with the prescribed initial conditions.

Hamiltonian Mechanics

In Hamiltonian mechanics, the principle of balance of mechanical energy in isothermal elastodynamical conservative systems reads as

$$\boxed{\frac{d}{dt}\mathscr{H}(\boldsymbol{\varphi}, \boldsymbol{\phi}) = \frac{d}{dt}\int_{\mathcal{B}} \mathcal{H}\left(\boldsymbol{\varphi}, \boldsymbol{\phi}, \mathbf{F}\right) dV - \frac{d}{dt}\int_{\partial\mathcal{B}_\sigma} \bar{\mathbf{T}} \cdot \boldsymbol{\varphi}\left(\mathbf{X}, t\right) dA = 0} \tag{3.69}$$

which leads to

$$\begin{aligned}\frac{d}{dt}\mathscr{H}(\boldsymbol{\varphi},\boldsymbol{\phi}) = \int_{\mathcal{B}} \left[\left(\dot{\boldsymbol{\phi}} + \frac{\partial \mathcal{H}}{\partial \boldsymbol{\varphi}} - \mathrm{DIV}\, \frac{\partial \mathcal{H}}{\partial \mathbf{F}} \right) \cdot \dot{\boldsymbol{\varphi}} + \left(\frac{\partial \mathcal{H}}{\partial \boldsymbol{\phi}} - \dot{\boldsymbol{\varphi}} \right) \cdot \dot{\boldsymbol{\phi}} \right] dV \\ - \int_{\partial \mathcal{B}_\sigma} \left[\bar{\mathbf{T}} - \frac{\partial \mathcal{H}}{\partial \mathbf{F}} \cdot \hat{\mathbf{N}} \right] \cdot \dot{\boldsymbol{\varphi}} dA = 0\end{aligned} \tag{3.70}$$

where the autonomous Hamiltonian and the Hamiltonian density function are defined in Eq. (3.50) and Eq. (3.51), respectively. Since both $\dot{\boldsymbol{\phi}}$ and $\dot{\boldsymbol{\varphi}}$ are *linearly independent*, we can readily obtain the strong form of the initial boundary-value problem for elastodynamical systems in terms of the autonomous Hamiltonian density function as summarized in Eq. (3.61). Note that the principle of balance of mechanical energy in Hamiltonian mechanics results in a two-field weighted residual form in $(\boldsymbol{\varphi}, \boldsymbol{\phi}) \in T^*\mathcal{C}$. Therefore, the strong form of the initial boundary-value problem for isothermal elastodynamical conservative systems can be obtained from the principle of balance of mechanical energy in the Hamiltonian mechanics framework.

Weak Form From Eq. (3.70), we can also obtain the weak form in the sense of the space as

$$\begin{aligned}\int_{\mathcal{B}} \dot{\boldsymbol{\phi}} \cdot \dot{\boldsymbol{\varphi}}\, dV + \int_{\mathcal{B}} \frac{\partial \mathcal{H}}{\partial \mathbf{F}} : \dot{\mathbf{F}}\, dV + \int_{\mathcal{B}} \frac{\partial \mathcal{H}}{\partial \boldsymbol{\varphi}} \cdot \dot{\boldsymbol{\varphi}}\, dV - \int_{\partial \mathcal{B}_\sigma} \bar{\mathbf{T}} \cdot \dot{\boldsymbol{\varphi}}\, dA = 0 \\ \int_{\mathcal{B}} \left(\dot{\boldsymbol{\varphi}} - \frac{\partial \mathcal{H}}{\partial \boldsymbol{\phi}} \right) \cdot \dot{\boldsymbol{\phi}}\, dV = 0\end{aligned} \tag{3.71}$$

with the prescribed initial conditions given in Eq. $(3.63)_3$ and Eq. $(3.63)_4$.

3.2.3 Spatial Discretizations by the Finite Element Method: Lagrangian and Total Energy Representations of Equation of Motion

Following the conventional Galerkin finite element method [21, 22], consider the admissible trial function $\boldsymbol{\varphi}^h$ represented by

$$\boldsymbol{\varphi}(\mathbf{X},t) \simeq \boldsymbol{\varphi}^h(\mathbf{X},t) = \mathbf{X} + \sum_{i=1}^{n_{\text{node}}} N_i(\mathbf{X})\,\mathbf{q}^i(t) \tag{3.72}$$

where n_{node} denotes the number of nodes, $N_i(\mathbf{X}) : \mathcal{B} \to \mathbb{R}$ denote the prescribed shape functions which satisfy the completeness condition $\sum_{i=1}^{n_{\text{node}}} N_i(\mathbf{X}) = 1$, and $\mathbf{q}^i(t) \in \mathbb{R}^3$ denote the nodal displacement vector associated with the i^{th} node at time $t \in \mathbb{I}$. For a set of nodal displacement vectors, we define the configuration manifold as

$$Q = \{\mathbf{q}(t) = (q^1(t), q^2(t), \cdots, q^{n_{\text{node}}}(t))^T : \mathbb{I} \to \mathbb{R}^{n_{\text{dof}}}\} \tag{3.73}$$

where $n_{\text{node}} = 3n_{\text{node}}$ denotes the number of degree of freedom. The trial function $\boldsymbol{\varphi}^h$ is of class $C^{m-1}(\mathcal{B})$ over the entire domain, and it belongs to the subspace

$$\mathcal{C}^h = \left\{ \begin{array}{l} \boldsymbol{\varphi}^h(\mathbf{X},t) : \mathcal{B}\times\mathbb{I} \to \mathbb{R}^3 | \boldsymbol{\varphi}^h \in W_2^m(\mathcal{B}) \subset C^{m-1}(\mathcal{B}), \text{ in } \mathcal{B} \\ \boldsymbol{\varphi}^h = \bar{\boldsymbol{\varphi}}, \text{ on } \partial\mathcal{B}_\varphi \end{array} \right\} \subset \mathcal{C} \tag{3.74}$$

From Eq. (3.72), the material velocity vector field may be approximated as

$$\mathbf{V}(\mathbf{X},t) \equiv \dot{\boldsymbol{\varphi}}(\mathbf{X},t) \simeq \dot{\boldsymbol{\varphi}}^h(\mathbf{X},t) = \sum_{i=1}^{n_{\text{node}}} \mathbf{N}_i(\mathbf{X})\,\dot{\mathbf{q}}^i(t) \in T_{\boldsymbol{\varphi}}\mathcal{C}^h \tag{3.75}$$

where $\dot{\mathbf{q}}(t) = (\dot{q}^1(t), \dot{q}^2(t), \cdots, \dot{q}^{n_{\text{node}}}(t))^T : \mathbb{I} \to T_{\mathbf{q}}Q$ is a set of the nodal velocity vector. Note that an arbitrary point $(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}})$ in the infinite dimensional tangent bundle $T\mathcal{C}$ is projected onto $(\boldsymbol{\varphi}^h, \dot{\boldsymbol{\varphi}}^h)$ in the finite dimensional tangent bundle $T\mathcal{C}^h \subset T\mathcal{C}$ by Eq. (3.72) and Eq. (3.75).

Discretization of the Weak Forms

Substituting Eq. (3.72) and Eq. (3.75) into the Lagrangian density function and mechanical energy density function, the Lagrangian and Total Energy versions of weak form given in Eq. (3.44) and Eq. (3.49) can be discretized in space as

$$\begin{aligned}&\int_{\mathcal{B}} \frac{d}{dt}\left(\frac{\partial \mathcal{L}^h}{\partial \dot{\boldsymbol{\varphi}}^h}\right) \cdot \delta \boldsymbol{\varphi}^h \, dV - \int_{\mathcal{B}} \frac{\partial \mathcal{L}^h}{\partial \mathbf{F}^h} : \delta \mathbf{F}^h \, dV \\ &- \int_{\mathcal{B}} \frac{\partial \mathcal{L}^h}{\partial \boldsymbol{\varphi}^h} \cdot \delta \boldsymbol{\varphi}^h \, dV - \int_{\partial \mathcal{B}_\sigma} \bar{\mathbf{T}} \cdot \delta \boldsymbol{\varphi}^h \, dA = 0\end{aligned} \tag{3.76}$$

and

$$\begin{aligned}&\int_{\mathcal{B}} \frac{d}{dt}\left(\frac{\partial \mathcal{E}^h}{\partial \dot{\boldsymbol{\varphi}}^h}\right) \cdot \delta \boldsymbol{\varphi}^h \, dV + \int_{\mathcal{B}} \frac{\partial \mathcal{E}^h}{\partial \mathbf{F}^h} : \delta \mathbf{F}^h \, dV \\ &+ \int_{\mathcal{B}} \frac{\partial \mathcal{E}^h}{\partial \boldsymbol{\varphi}^h} \cdot \delta \boldsymbol{\varphi}^h \, dV - \int_{\partial \mathcal{B}_\sigma} \bar{\mathbf{T}} \cdot \delta \boldsymbol{\varphi}^h \, dA = 0\end{aligned} \tag{3.77}$$

repsectively with the prescribed initial conditions, $\boldsymbol{\varphi}\left(\mathbf{X}, t_0\right) = \boldsymbol{\varphi}_0 = \mathbf{X}$ and $\dot{\boldsymbol{\varphi}}\left(\mathbf{X}, t_0\right) = \dot{\boldsymbol{\varphi}}_0 = \mathbf{V}_0$, where the space-discrete Lagrangian and mechanical energy density functions are given as

$$\mathcal{L}^h = \mathcal{L}(\boldsymbol{\varphi}^h, \dot{\boldsymbol{\varphi}}^h, \mathbf{F}^h) = \frac{1}{2}\rho_0 \dot{\boldsymbol{\varphi}}^h\left(\mathbf{X}, t\right) \cdot \dot{\boldsymbol{\varphi}}^h\left(\mathbf{X}, t\right) - \rho_0 \bar{\Psi}(\mathbf{F}^h) + \rho_0 \mathbf{B} \cdot \boldsymbol{\varphi}^h\left(\mathbf{X}, t\right) \tag{3.78}$$

$$\mathcal{E}^h = \mathcal{E}(\boldsymbol{\varphi}^h, \dot{\boldsymbol{\varphi}}^h, \mathbf{F}^h) = \frac{1}{2}\rho_0 \dot{\boldsymbol{\varphi}}^h\left(\mathbf{X}, t\right) \cdot \dot{\boldsymbol{\varphi}}^h\left(\mathbf{X}, t\right) + \rho_0 \bar{\Psi}(\mathbf{F}^h) - \rho_0 \mathbf{B} \cdot \boldsymbol{\varphi}^h\left(\mathbf{X}, t\right) \tag{3.79}$$

respectively. After some tedious calculations, both Eq. (3.76) and Eq. (3.77) lead to

$$\sum_{i=1}^{n_{\text{node}}} \left[\sum_{j=1}^{n_{\text{node}}} M_{ij} \ddot{\mathbf{q}} + \mathbf{F}_i^{\text{int}} - \mathbf{F}_i^{\text{ext}}\right] \cdot \delta \mathbf{q}^i = 0 \tag{3.80}$$

where $M_{ij} = \int_{\mathcal{B}} \rho_0 N_i N_j \, dV$ denotes the global symmetric mass matrix; and $\mathbf{F}_i^{\text{int}}$ and $\mathbf{F}_i^{\text{ext}}$ denote the internal and external force vectors associated with the i^{th}

node at time $t \in \mathbb{I}$ given by

$$\begin{aligned}
\mathbf{F}_i^{\text{int}} &= \int_{\mathcal{B}} \mathbf{P}^h \text{GRAD } N_i \, dV = \int_{\mathcal{B}} \mathbf{F}^h \cdot \bar{\mathbf{S}}^h(\mathbf{F}^h) \text{ GRAD } N_i \, dV \\
&= \sum_{j=1}^{n_{\text{node}}} \int_{\mathcal{B}} \text{GRAD } N_i \cdot \bar{\mathbf{S}}^h(\mathbf{F}^h) \text{GRAD } N_j dV \mathbf{q}^j \qquad (3.81) \\
\mathbf{F}_i^{\text{ext}} &= \int_{\mathcal{B}} \rho_0 N_i \mathbf{B} \, dV + \int_{\partial \mathcal{B}_\sigma} N_i \bar{\mathbf{T}} \, dA \qquad (3.82)
\end{aligned}$$

Note that $\bar{\mathbf{S}}^h(\mathbf{F}^h)$ denotes the stress response function for the space-discrete symmetric second Piola-Kirchhoff stress tensor field, i.e., $\mathbf{S}^h(\mathbf{X}, t) = \bar{\mathbf{S}}^h(\mathbf{F}^h)$, and the space-discrete deformation gradient tensor field is given as

$$\mathbf{F}^h(\mathbf{X}, t) = \text{GRAD } \boldsymbol{\varphi}^h(\mathbf{X}, t) = \mathbf{I} + \sum_{i=1}^{n_{\text{node}}} \mathbf{q}^i(t) \otimes \text{GRAD } N_i(\mathbf{X}) \qquad (3.83)$$

where $\mathbf{I}$ denotes the identity second-order tensor. Since $\delta\mathbf{q}$ is arbitrary, we obtain the initial-value problem for semi-discrete elastodynamical systems which yields Eq. (3.29).

Remark 4

1. *In contrast, the total energy version of weak form with the mechanical energy representation given in Eq.* (3.66) *(in the* ***differential calculus setting****) can be discretized in space via Eq.* (3.72) *and Eq.* (3.75) *as*

$$\int_{\mathcal{B}} \frac{d}{dt}\left(\frac{\partial \mathcal{E}^h}{\partial \dot{\boldsymbol{\varphi}}^h}\right) \cdot \dot{\boldsymbol{\varphi}}^h \, dV + \int_{\mathcal{B}} \frac{\partial \mathcal{E}^h}{\partial \mathbf{F}^h} : \dot{\mathbf{F}}^h \, dV + \int_{\mathcal{B}} \frac{\partial \mathcal{E}^h}{\partial \boldsymbol{\varphi}^h} \cdot \dot{\boldsymbol{\varphi}}^h \, dV - \int_{\partial \mathcal{B}_\sigma} \bar{\mathbf{T}} \cdot \dot{\boldsymbol{\varphi}}^h \, dA = 0 \qquad (3.84)$$

Likewise, in terms of the Lagrangian (in the ***differential calculus setting****), we have*

$$\int_{\mathcal{B}} \frac{d}{dt}\left(\frac{\partial \mathcal{L}^h}{\partial \dot{\boldsymbol{\varphi}}^h}\right) \cdot \dot{\boldsymbol{\varphi}}^h \, dV - \int_{\mathcal{B}} \frac{\partial \mathcal{L}^h}{\partial \mathbf{F}^h} : \dot{\mathbf{F}}^h \, dV - \int_{\mathcal{B}} \frac{\partial \mathcal{L}^h}{\partial \boldsymbol{\varphi}^h} \cdot \dot{\boldsymbol{\varphi}}^h \, dV - \int_{\partial \mathcal{B}_\sigma} \bar{\mathbf{T}} \cdot \dot{\boldsymbol{\varphi}}^h \, dA = 0 \qquad (3.85)$$

And both equations lead to

$$\sum_{i=1}^{n_{\text{node}}} \left[\sum_{j=1}^{n_{\text{node}}} M_{ij} \ddot{\mathbf{q}} + \mathbf{F}_i^{\text{int}} - \mathbf{F}_i^{\text{ext}} \right] \cdot \dot{\mathbf{q}}^i = 0 \qquad (3.86)$$

Since $\dot{\mathbf{q}}$ *is linearly independent, the initial-value problem for semi-discrete elastodynamical systems given in Eq.* (3.29) *is obtained.*

2. *The reader is referred referred to [7, 8] for alternate derivation via the space discrete finite element formalism.*

3.2.4 Spatial Discretizations by the Finite Element Method: Hamiltonian Representation of Equations of Motion

In order to project an arbitrary point $(\boldsymbol{\varphi}, \boldsymbol{\phi})$ in the infinite dimensional cotangent bundle $T^*\mathcal{C}$ onto $(\boldsymbol{\varphi}^h, \boldsymbol{\phi}^h)$ in the finite dimensional tangent bundle $T^*\mathcal{C}^h \subset T^*\mathcal{C}$, we employ

$$\begin{aligned} \boldsymbol{\varphi}(\mathbf{X}, t) \simeq \boldsymbol{\varphi}^h(\mathbf{X}, t) &= \mathbf{X} + \sum_{i=1}^{n_{\text{node}}} N_i(\mathbf{X})\, \mathbf{q}^i(t) \\ \boldsymbol{\phi}(\mathbf{X}, t) \simeq \boldsymbol{\phi}^h(\mathbf{X}, t) &= \sum_{i=1}^{n_{\text{node}}} N_i(\mathbf{X})\, \mathbf{p}^i(t) \end{aligned} \tag{3.87}$$

where $\boldsymbol{\varphi}^h$ and $\boldsymbol{\phi}^h$ denote the trial functions for the admissible configuration vector field and momentum vector field, respectively, and $\mathbf{p}(t) : \mathbb{I} \to T^*_{\mathbf{q}} Q$ is a set of nodal canonical momentum vector.

Discretization of the Weak Forms

Substituting Eq. $(3.87)_1$ and Eq. $(3.87)_2$ into the Hamiltonian density function, the Hamiltonian version of weak form given in Eq. (3.63) can be discretized in space as

$$\begin{aligned} \int_{\mathcal{B}} \dot{\boldsymbol{\phi}}^h \cdot \delta\boldsymbol{\varphi}^h \, dV + \int_{\mathcal{B}} \frac{\partial \mathcal{H}^h}{\partial \mathbf{F}^h} : \delta\mathbf{F}^h \, dV + \int_{\mathcal{B}} \frac{\partial \mathcal{H}^h}{\partial \boldsymbol{\varphi}^h} \cdot \delta\boldsymbol{\varphi}^h \, dV - \int_{\partial\mathcal{B}_\sigma} \bar{\mathbf{T}} \cdot \delta\boldsymbol{\varphi}^h \, dA = 0 \\ \int_{\mathcal{B}} \dot{\boldsymbol{\varphi}}^h \cdot \delta\boldsymbol{\phi}^h \, dV - \int_{\partial\mathcal{B}} \frac{\partial \mathcal{H}^h}{\partial \boldsymbol{\varphi}^h} \cdot \delta\boldsymbol{\phi}^h \, dV = 0 \end{aligned} \tag{3.88}$$

with the prescribed initial conditions, $\boldsymbol{\varphi}(\mathbf{X}, t_0) = \boldsymbol{\varphi}_0 = \mathbf{X}$ and $\boldsymbol{\phi}(\mathbf{X}, t_0) = \boldsymbol{\phi}_0$, where the space-discrete Hamiltonian density function is given as

$$\begin{aligned} \mathcal{H}^h = \mathcal{H}(\boldsymbol{\phi}^h, \boldsymbol{\varphi}^h, \mathbf{F}^h) &= \boldsymbol{\phi}^h \cdot \dot{\boldsymbol{\varphi}}^h - \mathcal{L}^h \\ &= \frac{1}{2\rho_0} \boldsymbol{\phi}^h(\mathbf{X}, t) \cdot \boldsymbol{\phi}^h(\mathbf{X}, t) \\ &\quad + \rho_0 \bar{\Psi}(\mathbf{F}^h) - \rho_0 \mathbf{B}(\mathbf{X}, t) \cdot \boldsymbol{\varphi}^h(\mathbf{X}, t) \end{aligned} \tag{3.89}$$

From Eq. (3.88), we can show

$$\sum_{i=1}^{n_{\text{node}}} \left[\left(\dot{\mathbf{p}}_i - \mathbf{F}_i^{\text{ext}} + \mathbf{F}_i^{\text{int}} \right) \cdot \delta \mathbf{q}^i - \left(\dot{\mathbf{q}}^i - \sum_{j=1}^{n_{\text{node}}} M_{ij}^{-1} \mathbf{p}_j \right) \cdot \delta \mathbf{p}_i \right] = 0 \tag{3.90}$$

Since both $\delta\mathbf{q}$ and $\delta\mathbf{p}$ are arbitrary, we obtain the initial-value problem for semi-discrete elastodynamical systems which reads

$$\boxed{\begin{aligned} \dot{\mathbf{p}}_i &= \mathbf{F}_i^{\text{ext}} - \mathbf{F}_i^{\text{int}} \\ \dot{\mathbf{q}}^i &= \sum_{j=1}^{n_{\text{node}}} M_{ij}^{-1} \mathbf{p}_j \\ \mathbf{q}(t_0) &= \mathbf{q}_0 \ \text{ and } \ \mathbf{p}(t_0) = \mathbf{p}_0 \end{aligned}} \tag{3.91}$$

where $\mathbf{q}_0 = (q^1(t_0), q^2(t_0), \cdots, q^{n_{\text{dof}}}(t_0))^T$ and $\mathbf{p}_0 = (p_1(t_0), p_2(t_0), \cdots, p_{n_{\text{dof}}}(t_0))^T$ are the prescribed nodal displacement and canonical momentum vectors at the initial time t_0.

Remark 5

1. *In contrast, the spatially discretized Hamiltonian version of weak form given in Eq.* (3.71) *(in the* ***differential calculus setting****), we have*

$$\begin{aligned} \int_{\mathcal{B}} \dot{\boldsymbol{\phi}}^h \cdot \dot{\boldsymbol{\varphi}}^h \, dV + \int_{\mathcal{B}} \frac{\partial \mathcal{H}^h}{\partial \mathbf{F}^h} : \dot{\mathbf{F}}^h \, dV + \int_{\mathcal{B}} \frac{\partial \mathcal{H}^{\text{h}}}{\partial \boldsymbol{\varphi}^h} \cdot \dot{\boldsymbol{\varphi}}^h \, dV - \int_{\partial \mathcal{B}_\sigma} \bar{\mathbf{T}} \cdot \dot{\boldsymbol{\varphi}}^h \, dA = 0 \\ \int_{\mathcal{B}} \left(\dot{\boldsymbol{\varphi}}^h - \frac{\partial \mathcal{H}^h}{\partial \boldsymbol{\phi}} \right) \cdot \dot{\boldsymbol{\phi}}^h \, dV = 0 \end{aligned} \tag{3.92}$$

which also lead to

$$\sum_{i=1}^{n_{\text{node}}} \left[\left(\dot{\mathbf{p}}_i - \mathbf{F}_i^{\text{ext}} + \mathbf{F}_i^{\text{int}} \right) \cdot \dot{\mathbf{q}}^i - \left(\dot{\mathbf{q}}^i - \sum_{j=1}^{n_{\text{node}}} M_{ij}^{-1} \mathbf{p}_j \right) \cdot \dot{\mathbf{p}}_i \right] = 0 \qquad (3.93)$$

Since both $\dot{\mathbf{q}}$ *and* $\dot{\mathbf{p}}$ *are linearly independent, we can obtain the initial-value problem for semi-discrete elastodynamical systems given in Eq.* (3.91)*.*

2. *The reader is referred to [7, 8] for alternate derivation via the space discrete finite element formalism.*

Chapter 4

Implicit Generalized Single Step Single Solve Framework and Family of Algorithms in Two- and Single-Field Forms: Linear Dynamical Systems

In this chapter, our goal is to temporally discretize the equation of motion in linear dynamical systems for a time interval $\mathbb{I} = [t_0, t_f]$, where t_0 and $t_f > 0$ are the initial time and the final time, respectively, split into subintervals, i.e., $\mathbb{I} = [t_0, t_f] = \bigcup_{n=0}^{f-1} [t_n, t_{n+1}]$ via a ***generalized time weighted residual methodology*** with focus upon deriving ***Implicit Generalized Single Step Single Solve (I-GSSSS) family of algorithms*** in single- and two-field forms. The total simulation time and the time step size are defined as $T := t_f - t_0$ and $\Delta t := t_{n+1} - t_n > 0$, respectively. Note that the time step size does not have to be constant. Linear dynamical algorithms are a necessary first step for subsequent extensions to nonlinear dynamical algorithms as described in a later chapter. They form the basis, and a sound theoretical basis is then established via

a new normalized time weighted residual methodology which describes how to enact the time discritization process for nonlinear dynamical problems leading to symplectic-momentum and energy-momentum algorithms by design.

4.1 I-GSSSS Framework and Family of Algorithms in Two-Field Form

The initial-value problem in the two-field form for linear dynamical systems consists of the linearized equation of motion with the kinematic constraint and the given initial conditions. That is, in general, we have

$$\boxed{\begin{aligned}
&\textbf{\textit{Balance Equation:}}\\
&\mathbf{M}\dot{\boldsymbol{\nu}}(t) + \mathbf{N}(\mathbf{q}(t), \boldsymbol{\nu}(t)) = \mathbf{f}(t) \quad \forall t \in \mathbb{I}\\
&\text{in which } \ \mathbf{N}(\mathbf{q}(t), \boldsymbol{\nu}(t)) = \mathbf{C}\boldsymbol{\nu}(t) + \mathbf{K}\mathbf{q}(t)\\
&\textbf{\textit{Kinematic constraint:}}\\
&\dot{\mathbf{q}}(t) = \boldsymbol{\nu}(t) \quad \forall t \in \mathbb{I}\\
&\textbf{\textit{Initial conditions:}}\\
&\mathbf{q}(t_0) = \mathbf{q}_0\\
&\boldsymbol{\nu}(t_0) = \boldsymbol{\nu}_0
\end{aligned}} \tag{4.1}$$

Note that the mass matrix, $\mathbf{M} \in \mathbb{R}^{n_{\text{dof}}} \times \mathbb{R}^{n_{\text{dof}}}$, is assumed constant. From Eq. (4.1), we have

$$\mathbf{M}\dot{\boldsymbol{\nu}}(t) + \mathbf{C}\boldsymbol{\nu}(t) + \mathbf{K}\mathbf{q}(t) - \mathbf{f}(t) = \mathbf{0} \quad \forall t \in \mathbb{I} \tag{4.2}$$

Approximating via an asymptotic series expansions for the dependent variables, $\mathbf{q}$, $\boldsymbol{\nu}$, and $\dot{\boldsymbol{\nu}}$, about time $t = t_{n+\alpha} := (1-\alpha)t_n + \alpha t_{n+1} \in [t_n, t_{n+1}]$ for $\alpha \in [0, 1]$

yields

$$\begin{aligned}
\mathbf{q}(t_{n+\alpha}) &\cong \mathbf{q}(t_n) + \Lambda_1 \dot{\mathbf{q}}(t_n)[t_{n+\alpha} - t_n] \\
&\quad + \Lambda_2 \frac{\dot{\mathbf{q}}(t_{n+1}) - \dot{\mathbf{q}}(t_n)}{\Delta t}[t_{n+\alpha} - t_n]^2 \qquad (4.3) \\
\boldsymbol{\nu}(t_{n+\alpha}) &\cong \boldsymbol{\nu}(t_n) + \Lambda_4 \frac{\boldsymbol{\nu}(t_{n+1}) - \boldsymbol{\nu}(t_n)}{\Delta t}[t_{n+\alpha} - t_n] =: \hat{\boldsymbol{\nu}} \qquad (4.4) \\
\dot{\boldsymbol{\nu}}(t_{n+\alpha}) &\cong \frac{\boldsymbol{\nu}(t_{n+1}) - \boldsymbol{\nu}(t_n)}{\Delta t} =: \hat{\dot{\boldsymbol{\nu}}} \qquad (4.5)
\end{aligned}$$

respectively with algorithmic parameters $\Lambda_i \in \mathbb{R}$ $(i = 1, 2, 3)$. Because of the kinematic constraints at time t_n and t_{n+1}, i.e., $\dot{\mathbf{q}}(t_n) = \boldsymbol{\nu}(t_n)$ and $\dot{\mathbf{q}}(t_{n+1}) = \boldsymbol{\nu}(t_{n+1})$, respectively, Eq. (4.3) may be written as

$$\begin{aligned}
\mathbf{q}(t_{n+\alpha}) \cong &\mathbf{q}(t_n) + \Lambda_1 \boldsymbol{\nu}(t_n)[t_{n+\alpha} - t_n] \\
&+ \Lambda_2 \frac{\boldsymbol{\nu}(t_{n+1}) - \boldsymbol{\nu}(t_n)}{\Delta t}[t_{n+\alpha} - t_n]^2 =: \hat{\mathbf{q}}
\end{aligned} \qquad (4.6)$$

For the time-dependent external force vector, we linearly approximate within a time step $[t_n, t_{n+1}]$ as

$$\mathbf{f}(t_{n+\alpha}) \cong \mathbf{f}(t_n) + \frac{\mathbf{f}(t_{n+1}) - \mathbf{f}(t_n)}{\Delta t}[t_{n+\alpha} - t_n] =: \hat{\mathbf{f}} \qquad (4.7)$$

Substituting Eq. (4.4) - Eq. (4.6) into (4.7), the residual, $\mathbf{r} \neq \mathbf{0}$ in general, is defined as

$$\mathbf{r} := \mathbf{M}\hat{\dot{\boldsymbol{\nu}}} + \mathbf{C}\hat{\boldsymbol{\nu}} + \mathbf{K}\hat{\mathbf{q}} - \hat{\mathbf{f}} \qquad (4.8)$$

or

$$\begin{aligned}
\mathbf{r} :=&\mathbf{M}\frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t} + \mathbf{C}\left[\boldsymbol{\nu}_n + \Lambda_4 \frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t}\tau\right] \\
&+ \mathbf{K}\left[\mathbf{q}_n + \Lambda_1 \boldsymbol{\nu}_n \tau + \Lambda_2 \frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t}\tau^2\right] - \left[\mathbf{f}_n + \frac{\mathbf{f}_{n+1} - \mathbf{f}_n}{\Delta t}\tau\right]
\end{aligned} \qquad (4.9)$$

where $\tau := t_{n+\alpha} - t_n = \alpha \Delta t$; and $\square_{n+1} \approx \square(t_{n+1})$ and $\square_n \approx \square(t_n)$. Employing a ***generalized time weighted residual methodology*** within the time step for the residual $\mathbf{r}$, we have

$$\frac{\int_0^{\Delta t} W \mathbf{r} d\tau}{\int_0^{\Delta t} W d\tau} = \frac{\int_0^{\Delta t} W \left[\mathbf{M}\hat{\dot{\boldsymbol{\nu}}} + \mathbf{C}\hat{\boldsymbol{\nu}} + \mathbf{K}\hat{\mathbf{q}} - \hat{\mathbf{f}}\right] d\tau}{\int_0^{\Delta t} W d\tau} \cong \mathbf{0} \qquad (4.10)$$

or

$$\tilde{\mathbf{r}} = \mathbf{M}\tilde{\dot{\boldsymbol{\nu}}} + \mathbf{C}\tilde{\boldsymbol{\nu}} + \mathbf{K}\tilde{\mathbf{q}} - \tilde{\mathbf{f}} \cong \mathbf{0} \tag{4.11}$$

where the algorithmic unknowns are defined as

$$\tilde{\square} := \frac{\int_0^{\Delta t} W \square d\tau}{\int_0^{\Delta t} W d\tau} \tag{4.12}$$

Note that $W \in \mathbb{R}$ denotes a degenerated scalar polynomial weighting time field [7, 8] defined as

$$W(\tau) \cong w_0 + w_1 \frac{\tau}{\Delta t} + w_2 \left(\frac{\tau}{\Delta t}\right)^2 \tag{4.13}$$

with $w_i \in \mathbb{R}$ (for $i = 0, 1, 2$). We normally set $w_0 = 1$ ($w_0 \neq 0$ in general); see [28] for details. Replacing

$$\begin{aligned} \frac{\int_0^{\Delta t} \frac{\tau}{\Delta t} W d\tau}{\int_0^{\Delta t} W d\tau} &= \frac{6 + 4w_1 + 3w_2}{12 + 6w_1 + 4w_2}, \text{ and} \\ \frac{\int_0^{\Delta t} \left(\frac{\tau}{\Delta t}\right)^2 W d\tau}{\int_0^{\Delta t} W d\tau} &= \frac{20 + 15w_1 + 12w_2}{60 + 30w_1 + 20w_2} \end{aligned} \tag{4.14}$$

with algorithmic parameters, W_1 and W_2, respectively, we have

$$\tilde{\mathbf{q}} = \mathbf{q}_n + \Delta t W_1 \Lambda_1 \boldsymbol{\nu}_n + \Delta t W_2 \Lambda_2 (\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n) \tag{4.15}$$

$$\tilde{\boldsymbol{\nu}} = \boldsymbol{\nu}_n + W_1 \Lambda_4 (\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n) \tag{4.16}$$

$$\tilde{\dot{\boldsymbol{\nu}}} = \frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t} \tag{4.17}$$

and

$$\tilde{\mathbf{f}} = \mathbf{f}_n + W_1 (\mathbf{f}_{n+1} - \mathbf{f}_n) \tag{4.18}$$

Notice that the algorithmic external force vector, $\tilde{\mathbf{f}}$, satisfies $\mathbf{f}(t_{n+W_1}) = \tilde{\mathbf{f}} + \mathcal{O}(\Delta t^2)$ if $W_1 \neq 0$ or $W_1 \neq 1$. The corresponding updates are obtained by setting $\alpha = 1$ in Eq. (4.3) and Eq. (4.4) and replacing Λ_1 and Λ_2 with new algorithmic parameters, $\lambda_1 \in \mathbb{R}$ and $\lambda_2 \in \mathbb{R}$, respectively. That is,

$$\begin{aligned} \mathbf{q}_{n+1} &= \mathbf{q}_n + \Delta t \lambda_1 \boldsymbol{\nu}_n + \Delta t \lambda_2 (\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n) \\ \boldsymbol{\nu}_{n+1} &= \boldsymbol{\nu}_n + (\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n) \end{aligned} \tag{4.19}$$

Note that the update, Eq. (4.19), can be considered as the discrete form of the kinematic constraint, i.e.,

$$\frac{\mathbf{q}_{n+1}-\mathbf{q}_n}{\Delta t}=\lambda_1\boldsymbol{\nu}_n+\lambda_2(\boldsymbol{\nu}_{n+1}-\boldsymbol{\nu}_n) \tag{4.20}$$

Rearranging the representation of the algorithm discussed above leads to the general form of the implicit GSSSS framework and family of algorithms in two-field form (or the ***two-field form I-GSSSS framework and family of algorithms***) as

V-form

Integrator:

$$\left[\frac{1}{\Delta t}\mathbf{M}+W_1\Lambda_4\mathbf{C}+\Delta t W_2\Lambda_2\mathbf{K}\right]\Delta\boldsymbol{\nu}$$

$$=-\mathbf{C}\boldsymbol{\nu}_n-\mathbf{K}\left[\mathbf{q}_n+\Delta t W_1\Lambda_1\boldsymbol{\nu}_n\right]+(1-W_1)\mathbf{f}_n+W_1\mathbf{f}_{n+1}$$

Updates:

$$\mathbf{q}_{n+1}=\mathbf{q}_n+\Delta t\lambda_1\boldsymbol{\nu}_n+\Delta t\lambda_2\Delta\boldsymbol{\nu}$$

$$\boldsymbol{\nu}_{n+1}=\boldsymbol{\nu}_n+\Delta\boldsymbol{\nu}$$

Initial conditions:

$$\mathbf{q}(t_0)=\mathbf{q}_0$$

$$\boldsymbol{\nu}(t_0)=\boldsymbol{\nu}_0 \tag{4.21}$$

where the $\boldsymbol{\nu}$-increment $\Delta\boldsymbol{\nu} := \boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n$ is the primary unknown in the ***V-form*** representation of the two-field form I-GSSSS framework. Or equivalently,

D-form

Integrator:

$$\begin{aligned}
&\left[\frac{1}{\Delta t^2\lambda_2}\mathbf{M} + \frac{W_1\Lambda_4}{\Delta t\lambda_2}\mathbf{C} + \frac{W_2\Lambda_2}{\lambda_2}\mathbf{K}\right]\Delta\mathbf{q} \\
&= \frac{\lambda_1}{\Delta t\lambda_2}\mathbf{M}\boldsymbol{\nu}_n - \left(1 - W_1\Lambda_4\frac{\lambda_1}{\lambda_2}\right)\mathbf{C}\boldsymbol{\nu}_n \\
&- \mathbf{K}\left[\mathbf{q}_n + \Delta t\left(W_1\Lambda_1 - W_2\Lambda_2\frac{\lambda_1}{\lambda_2}\right)\boldsymbol{\nu}_n\right] + (1 - W_1)\mathbf{f}_n + W_1\mathbf{f}_{n+1}
\end{aligned} \tag{4.22}$$

Updates:

$$\boldsymbol{\nu}_{n+1} = \left(1 - \frac{\lambda_1}{\lambda_2}\right)\boldsymbol{\nu}_n + \frac{1}{\Delta t\lambda_2}\Delta\mathbf{q}$$

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \Delta\mathbf{q}$$

Initial conditions:

$$\mathbf{q}(t_0) = \mathbf{q}_0$$

$$\boldsymbol{\nu}(t_0) = \boldsymbol{\nu}_0$$

where the $\mathbf{q}$-increment $\Delta\mathbf{q} := \mathbf{q}_{n+1} - \mathbf{q}_n$ is the primary unknown in the ***D-form*** representation of the two-field form I-GSSSS framework.

The complete algorithmic characteristics can be studied by analyzing a single-degree-of-freedom (SDOF) situation by virtue of the mode superposition. The reduction of the algorithm shown in Eq. (4.21) to a SDOF problem leads to

$$\begin{aligned}
&\left[1 + 2W_1\Lambda_4\xi\omega\Delta t + \Lambda_2 W_2\omega^2\Delta t^2\right]\Delta\nu \\
&= -2\xi\omega\Delta t q_n - \omega^2\Delta t\left[q_n + \Delta t W_1\Lambda_1\nu_n\right] + \Delta t g_{n+W_1} \\
&q_{n+1} = q_n + \Delta t\lambda_1\nu_n + \Delta t\lambda_2\Delta\nu \\
&\nu_{n+1} = \nu_n + \Delta\nu
\end{aligned} \tag{4.23}$$

with the given initial conditions $q(t_0) = q_0$ and $\nu(t_0) = \nu_0$, where ξ and ω denote the physical damping ratio and eigen-frequency for each degree of freedom, respectively. The above SDOF representation of the two-field form I-GSSSS family

of algorithms can be cast into the following form:

$$\hat{\mathbf{y}}_{n+1} = \hat{\mathbf{A}}\hat{\mathbf{y}}_n + \hat{\mathbf{L}}_{n+W_1} \tag{4.24}$$

where the state vectors, the 2×2 amplification $\hat{\mathbf{A}}$ and the load vector $\hat{\mathbf{L}}_{n+W_1}$ are given by

$$\begin{aligned} \hat{\mathbf{y}}_{n+1} &= \begin{Bmatrix} q_{n+1} \\ \nu_{n+1} \end{Bmatrix}, \quad \hat{\mathbf{y}}_n = \begin{Bmatrix} q_n \\ \nu_n \end{Bmatrix} \\ \hat{\mathbf{A}} &= \begin{bmatrix} 1 + \lambda_2\alpha_1 & (\lambda_1 + \lambda_2\alpha_2)\Delta t \\ \frac{\alpha_1}{\Delta t} & 1 + \alpha_2 \end{bmatrix} \\ \hat{\mathbf{L}}_{n+W_1} &= \frac{1}{D}\left[(1 - W_1)g_n + W_1 g_{n+1}\right] \begin{pmatrix} \lambda_2\Delta t^2 \\ \Delta t \end{pmatrix} \end{aligned} \tag{4.25}$$

respectively, where

$$\begin{aligned} \alpha_1 &= -\frac{\Omega^2}{D}, \quad \alpha_2 = -\frac{2\xi\Omega + W_1\Lambda_1\Omega^2}{D} \\ D &= 1 + 2\xi\Omega W_1\Lambda_4 + \Omega^2 W_2\Lambda_2 \\ \Omega &= \omega\Delta t \end{aligned} \tag{4.26}$$

Or, it may be more convenient to define the state vectors as $\mathbf{y}_{n+1} = \{q_{n+1}, \Delta t\nu_{n+1}\}^T$ and $\mathbf{y}_n = \{q_n, \Delta t\nu_n\}^T$ such that

$$\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n + \mathbf{L}_{n+W_1} \tag{4.27}$$

where the amplification matrix and the load vector can be written as

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} 1 + \lambda_2\alpha_1 & \lambda_1 + \lambda_2\alpha_2 \\ \alpha_1 & 1 + \alpha_2 \end{bmatrix} \\ \mathbf{L}_{n+W_1} &= \frac{\Delta t^2}{D}\left[(1 - W_1)g_n + W_1 g_{n+1}\right] \begin{pmatrix} \lambda_2 \\ 1 \end{pmatrix} \end{aligned} \tag{4.28}$$

respectively. For both cases, the principal invariants of the amplification matrix

are given as

$$\begin{aligned} A_1 &= \mathrm{tr}(\hat{\mathbf{A}}) = \mathrm{tr}(\mathbf{A}) \\ &= \frac{2 - 2(1 - 2W_1\Lambda_4)\xi\Omega - (W_1\Lambda_1 + \lambda_2 - 2W_2\Lambda_2)\Omega^2}{D} \end{aligned} \tag{4.29}$$

$$\begin{aligned} A_2 &= \det(\hat{\mathbf{A}}) = \det(\mathbf{A}) \\ &= \frac{1 - 2(1 - W_1\Lambda_4)\xi\Omega + (\lambda_1 - W_1\Lambda_1 - \lambda_2 + W_2\Lambda_2)\Omega^2}{D} \end{aligned} \tag{4.30}$$

The characteristic polynomial of the amplification matrix is given by

$$\det(\zeta\mathbf{I}_2 - \mathbf{A}) = \zeta^2 - A_1\zeta + A_2 = 0 \tag{4.31}$$

where ζ denotes the eigenvalues of $\mathbf{A}$ and $\mathbf{I}_2$ denotes the 2×2 identity matrix. Employing the Cayley-Hamilton theorem, Eq. (4.31) can be written as

$$\mathbf{A}^2 - A_1\mathbf{A} + A_2\mathbf{I}_2 = \mathbf{0} \tag{4.32}$$

Therefore, for the homogeneous case, i.e., $\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n$, we have

$$\begin{aligned} \mathbf{y}_{n+2} &= \mathbf{A}\mathbf{y}_{n+1} = \mathbf{A}^2\mathbf{y}_n \\ &= (A_1\mathbf{A} - A_2\mathbf{I}_2)\mathbf{y}_n \\ &= A_1\mathbf{A}\mathbf{y}_n - A_2\mathbf{y}_n \end{aligned} \tag{4.33}$$

Noting $\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n$, we readily obtain the difference equation which takes the form

$$\mathbf{y}_{n+2} - A_1\mathbf{y}_{n+1} + A_2\mathbf{y}_n = \mathbf{0} \tag{4.34}$$

For the non-homogeneous case, i.e., $\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n + \mathbf{L}_{n+W_1}$, the difference equation yields

$$\mathbf{y}_{n+2} - A_1\mathbf{y}_{n+1} + A_2\mathbf{y}_n = (\mathbf{A} - A_1\mathbf{I}_2)\,\mathbf{L}_{n+W_1} + \mathbf{L}_{n+1+W_1} \tag{4.35}$$

where

$$\mathbf{L}_{n+1+W_1} = \frac{\Delta t^2}{D}\left[(1 - W_1)g_{n+1} + W_1 g_{n+2}\right]\begin{pmatrix}\lambda_2 \\ 1\end{pmatrix} \tag{4.36}$$

Order of Time Accuracy

Consider the homogeneous case for simplicity. We define the local truncation error of the linear multi-step representation for q from the difference equation given by Eq. (4.34) as

$$\tau_q^{(1)}(\Delta t) := \frac{1}{\Delta t^2}\left[q(t_{n+2}) - A_1 q(t_{n+1}) + A_2 q(t_n)\right] \tag{4.37}$$

where $q(t_{n+2})$, $q(t_{n+1})$, and $q(t_n)$ denote the exact solutions at time t_{n+2}, t_{n+1}, and t_n, respectively, i.e., $q(t_{n+2}) \approx q_{n+2}$, $q(t_{n+1}) \approx q_{n+1}$, and $q(t_n) \approx q_n$. Substituting the Taylor series expansions of $q(t_{n+2})$ and $q(t_{n+1})$ at time t_n,

$$q(t_{n+2}) = q(t_n) + 2\Delta t\dot{q}(t_n) + 2\Delta t^2\ddot{q}(t_n) + \mathcal{O}(\Delta t^3) \tag{4.38}$$

$$q(t_{n+1}) = q(t_n) + \Delta t\dot{q}(t_n) + \frac{\Delta t^2}{2}\ddot{q}(t_n) + \mathcal{O}(\Delta t^3) \tag{4.39}$$

into Eq. (4.37) yields

$$\begin{aligned}\tau_q^{(1)}(\Delta t) = {} & \frac{1}{D}\left[\ddot{q}(t_n) + 2\omega\xi\dot{q}(t_n) + \lambda_1\omega^2 q(t_n)\right] \\ & + \frac{\Delta t}{D}\left[\dddot{q}(t_n) + (1 + 2W_1\Lambda_4)\omega\xi\ddot{q}(t_n) + (\lambda_2 + W_1\Lambda_1)\omega^2\dot{q}(t_n)\right] \\ & + \mathcal{O}(\Delta t^2)\end{aligned} \tag{4.40}$$

or

$$\begin{aligned}\tau_q^{(1)}(\Delta t) = {} & \frac{1}{D}\left[\dot{\nu}(t_n) + 2\omega\xi\nu(t_n) + \lambda_1\omega^2 q(t_n)\right] \\ & + \frac{\Delta t}{D}\left[\ddot{\nu}(t_n) + (1 + 2W_1\Lambda_4)\omega\xi\dot{\nu}(t_n) + (\lambda_2 + W_1\Lambda_1)\omega^2\nu(t_n)\right] \\ & + \mathcal{O}(\Delta t^2)\end{aligned} \tag{4.41}$$

due to the kinematic constraint. Therefore, $\tau_q^{(1)}(\Delta t) = \mathcal{O}(\Delta t)$ is obtained if $\lambda_1 = 1$ and all other algorithmic parameters are arbitrary. And, $\tau_q^{(1)}(\Delta t) = \mathcal{O}(\Delta t^2)$ is obtained if

$$\begin{aligned}\lambda_1 &= 1 \\ W_1\Lambda_4 &= \frac{1}{2} \\ \lambda_2 + W_1\Lambda_1 &= 1\end{aligned} \tag{4.42}$$

However, the conditions given in Eq. (4.42) are unfortunately not the necessary and sufficient conditions for the second-order time accuracy of the algorithms. We define the local truncation error for q from $\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n$ as

$$\tau_q^{(2)}(\Delta t) := q(t_{n+1}) - A_{11}q(t_n) - \Delta t A_{12}\nu(t_n) \tag{4.43}$$

Note that $D = \mathcal{O}(1)$. Substituting Eq. (4.39) into Eq. (5.28), we get

$$\begin{aligned}\tau_q^{(2)}(\Delta t) = \frac{\Delta t}{D}(1-\lambda_1)\nu(t_n) \\ + \frac{\Delta t^2}{2D}[\dot{\nu}(t_n) + 4\left(\lambda_2 + (1-\lambda_1)W_1\Lambda_4\right)\xi\omega\nu(t_n) \\ + 2\lambda_2\omega^2 q(t_n)] + \mathcal{O}(\Delta t^3)\end{aligned} \tag{4.44}$$

Therefore, $\tau_q^{(2)}(\Delta t) = \mathcal{O}(\Delta t)$ is obtained for all arbitrary algorithmic parameters; $\tau_q^{(2)}(\Delta t) = \mathcal{O}(\Delta t^2)$ is obtained if $\lambda_1 = 1$ and all other algorithmic parameters are arbitrary; and $\tau_q^{(2)}(\Delta t) = \mathcal{O}(\Delta t^3)$ is obtained if

$$\lambda_1 = 1 \ \text{ and } \ \lambda_2 = \frac{1}{2} \tag{4.45}$$

and all other algorithmic parameters are arbitrary. To guarantee the second-order time accuracy of the various algorithms, we must have both $\tau_q^{(1)}(\Delta t) = \mathcal{O}(\Delta t^2)$ and $\tau_q^{(2)}(\Delta t) = \mathcal{O}(\Delta t^3)$. Hence,

$$\boxed{\begin{aligned}&\lambda_1 = 1, \ \ \lambda_2 = \frac{1}{2}, \ \ W_1\Lambda_1 = \frac{1}{2}, \ \text{ and } \ W_1\Lambda_4 = \frac{1}{2} \\ &\qquad\qquad\qquad\qquad W_2, \Lambda_2 : \ \text{ arbitrary} \\ &\qquad\qquad\qquad\qquad \textbf{(for homogeneous case)}\end{aligned}} \tag{4.46}$$

The necessary and sufficient conditions for the second-order time accuracy of the algorithms given above can be actually obtained without also using the local truncation error of the multi-step form, $\tau_q^{(1)}(\Delta t) = \mathcal{O}(\Delta t^2)$. Define the local truncation error for ν from $\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n$ as

$$\tau_\nu^{(2)}(\Delta t) := \nu(t_{n+1}) - \frac{A_{21}}{\Delta t}q(t_n) - A_{22}\nu(t_n) \tag{4.47}$$

Substituting the Taylor series expansion of $\nu(t_{n+1})$ at time t_n,

$$\nu(t_{n+1}) = \nu(t_n) + \Delta t\dot{\nu}(t_n) + \frac{\Delta t^2}{2}\ddot{\nu}(t_n) + \mathcal{O}(\Delta t^3) \tag{4.48}$$

we get

$$\tau_\nu^{(2)}(\Delta t) = \frac{\Delta t^2}{2D}\left[\dot{\nu}(t_n) + 4W_1\Lambda_4\xi\omega\nu(t_n) + 2W_1\Lambda_1\omega^2\boldsymbol{\nu}(t_n)\right] + \mathcal{O}(\Delta t^3) \tag{4.49}$$

Therefore, $\tau_\nu^{(2)}(\Delta t) = \mathcal{O}(\Delta t^2)$ is obtained for all arbitrary algorithmic parameters; and $\tau_\nu^{(2)}(\Delta t) = \mathcal{O}(\Delta t^3)$ is obtained if

$$W_1\Lambda_1 = \frac{1}{2} \text{ and } W_1\Lambda_4 = \frac{1}{2} \tag{4.50}$$

and all other algorithmic parameters are arbitrary. Hence, the second-order time accuracy conditions given in Eq. (4.46) are obtained from $\tau_q^{(2)}(\Delta t) = \mathcal{O}(\Delta t^3)$ and $\tau_\nu^{(2)}(\Delta t) = \mathcal{O}(\Delta t^3)$, i.e., Eq. (4.45) and Eq. (4.50).

For the non-homogeneous case, consider the local truncation error vector, $\boldsymbol{\tau}(\Delta t) = \{\tau_q(\Delta t), \tau_\nu(\Delta t)\}^T$, defined by replacing $\hat{\mathbf{y}}_{n+1}$ and $\hat{\mathbf{y}}_n$ in Eq.(4.24) with the exact solutions $\hat{\mathbf{y}}(t_{n+1})$ and $\hat{\mathbf{y}}(t_n)$; that is,

$$\hat{\mathbf{y}}(t_{n+1}) - \hat{\mathbf{A}}\hat{\mathbf{y}}(t_n) - \hat{\mathbf{L}}_{n+W_1} = \Delta t\boldsymbol{\tau}(\Delta t) \tag{4.51}$$

where the amplification matrix and the load vector are given in Eq. (4.25); therefore,

$$\begin{aligned}\Delta t\begin{Bmatrix}\tau_q(\Delta t)\\ \tau_\nu(\Delta t)\end{Bmatrix} = \begin{Bmatrix}q(t_{n+1})\\ \nu(t_{n+1})\end{Bmatrix} - \begin{bmatrix}1+\lambda_2\alpha_1 & (\lambda_1+\lambda_2\alpha_2)\Delta t\\ \frac{\alpha_1}{\Delta t} & 1+\alpha_2\end{bmatrix}\begin{Bmatrix}q(t_n)\\ \nu(t_n)\end{Bmatrix}\\ - \frac{1}{D}\begin{Bmatrix}\lambda_2\Delta t^2\left[(1-W_1)g(t_n) + W_1 g(t_{n+1})\right]\\ \Delta t\left[(1-W_1)g(t_n) + W_1 g(t_{n+1})\right]\end{Bmatrix}\end{aligned} \tag{4.52}$$

If $\| \boldsymbol{\tau}(\Delta t) \| = \mathcal{O}(\Delta t^k)\ \forall t \in \mathbb{I}$ where $k > 0$ denotes the order of time accuracy (the rate of convergence), the single-step algorithms are consistent. If $\| \boldsymbol{\tau}(\Delta t) \| = \mathcal{O}(\Delta t^2)\ \forall t \in \mathbb{I}$, the order of time accuracy of the selected algorithm is 2. To check the order of time accuracy of Eq.(4.24), expand $q(t_{n+1})$, $\nu(t_{n+1})$ and $g(t_{n+1})$

about time t_n by the Taylor series expansions. After some tedious calculations, we obtain the following expressions of the local truncation errors for the position and velocity:

$$\begin{aligned}\tau_q(\Delta t) &= \frac{(1-\lambda_1)}{D}\dot{q}(t_n) \\ &\quad + \frac{\Delta t}{D}\left(\frac{1}{2}\ddot{q}(t_n) + 2\left[\lambda_2 + (1-\lambda_1)W_1\Lambda_4\right]\xi\omega\dot{q}(t_n) + \lambda_2\omega^2 q(t_n) - \lambda_2 g(t_n)\right) \\ &\quad + \mathcal{O}(\Delta t^2) \qquad (4.53) \\ \tau_\nu(\Delta t) &= \frac{\Delta t}{D}\left(\frac{1}{2}\ddot{\nu}(t_n) + 2W_1\Lambda_4\xi\omega\dot{\nu}(t_n) + W_1\Lambda_1\omega^2\nu(t_n) - W_1 g(t_n)\right) \\ &\quad + \mathcal{O}(\Delta t^2) \qquad (4.54)\end{aligned}$$

Therefore, for the second-order time accuracy, i.e., $\tau_q(\Delta t) = \mathcal{O}(\Delta t^2)$ and $\tau_\nu(\Delta t) = \mathcal{O}(\Delta t^2)$, we must have

$$1 - \lambda_1 = 0 \ \text{ and } \ \frac{1}{2} = \lambda_2 + (1-\lambda_1)W_1\Lambda_4 = \lambda_2 \qquad (4.55)$$

for $\tau_q(\Delta t) = \mathcal{O}(\Delta t^2)$, and

$$\frac{1}{2} = W_1\Lambda_4 = W_1\Lambda_1 = W_1 \qquad (4.56)$$

for $\tau_\nu(\Delta t) = \mathcal{O}(\Delta t^2)$. Hence,

$$\boxed{\begin{array}{r}\lambda_1 = 1, \ \ \lambda_2 = \frac{1}{2}, \ \ \Lambda_1 = 1, \ \ \Lambda_4 = 1, \text{ and } \ W_1 = \frac{1}{2} \\ W_2, \Lambda_2 : \ \text{ arbitrary} \\ \textbf{(for non-homogeneous case)}\end{array}} \qquad (4.57)$$

Overshoot

Consider the SDOF homogeneous equation, $\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n$. When an unconditionally stable algorithm is applied to multi-degree of freedom problems, we often tend to have the presence of the so-called high-frequency phenomenon, i.e., large values of Ω can occur. In order to investigate the overshooting behavior of the algorithm,

we approximate the first time step of the scheme with $\mathbf{A}_\infty := \lim_{\Omega\to\infty} \mathbf{A}$. For the first time step from $n = 0$ to $n = 1$,

$$\mathbf{y}_1 \simeq \mathbf{A}_\infty \mathbf{y}_0 \tag{4.58}$$

where $\mathbf{y}_1 = \{q_1, \Delta t\nu_1\}^T$ and $\mathbf{y}_0 = \{q_0, \Delta t\nu_0\}^T = \{q(t_0), \Delta t\nu(t_0)\}^T$; and the amplification matrix at the high-frequency limit is given by

$$\mathbf{A}_\infty = \lim_{\Omega\to\infty} \mathbf{A} = \begin{bmatrix} 1 - \frac{\lambda_2}{W_2\Lambda_2} & \lambda_1 - \lambda_2 \frac{W_1\Lambda_1}{W_2\Lambda_2} \\ -\frac{1}{W_2\Lambda_2} & 1 - \frac{W_1\Lambda_1}{W_2\Lambda_2} \end{bmatrix} \tag{4.59}$$

Therefore, we readily have

$$q_1 \simeq \left(1 - \frac{\lambda_2}{W_2\Lambda_2}\right) q_0 + \left(\lambda_1 - \lambda_2 \frac{W_1\Lambda_1}{W_2\Lambda_2}\right) \Delta t\nu_0 \tag{4.60}$$

$$\nu_1 \simeq -\frac{1}{W_2\Lambda_2\Delta t} q_0 + \left(1 - \frac{W_1\Lambda_1}{W_2\Lambda_2}\right) \nu_0 \tag{4.61}$$

Eq. (4.60) and Eq. (4.61) show there exists no overshoot in both configuration and velocity for any conditions of the parameters.

Spectral Analysis

The eigenvalues of the amplification matrix $\mathbf{A}$ are given as

$$\zeta_{1,2} = \frac{1}{D}\left[1 - \Omega\left((1 - 2W_1\Lambda_4)\xi + \frac{W_1\Lambda_1 + \lambda_2 - 2W_2\Lambda_2}{2}\Omega \pm \frac{\sqrt{\delta}}{2}\right)\right] \tag{4.62}$$

where

$$\delta = \left[(W_1\Lambda_1 + \lambda_2)^2 - 4\lambda_1 W_2\Lambda_2\right]\Omega^2 + 4\xi\left(W_1\Lambda_1 + \lambda_2 - 2\lambda_1 W_1\Lambda_4\right)\Omega + 4(\xi^2 - \lambda_1) \tag{4.63}$$

Therefore, the bifurcation sample frequency, Ω_{bif}, at which two complex conjugate eigenvalues turn into two real eigenvalues is given by

$$\Omega_{\text{bif}} = \left|\frac{-2\xi\left(\lambda_2 + W_1\Lambda_1 - 2\lambda_1 W_1\Lambda_4\right) \pm \sqrt{\upsilon}}{(W_1\Lambda_1 + \lambda_2)^2 - 4\lambda_1 W_2\Lambda_2}\right| \tag{4.64}$$

where

$$\upsilon = \lambda_1 \left[(W_1\Lambda_1 + \lambda_2)^2 - 4W_1\Lambda_4 \left(\lambda_2 + W_1\Lambda_1 - \lambda_1 W_1\Lambda_4\right)\xi^2 + 4W_2\Lambda_2(\xi^2 - \lambda_1) \right] \tag{4.65}$$

which shows no bifurcation occurs if and only if $(W_1\Lambda_1 + \lambda_2)^2 = 4\lambda_1 W_2\Lambda_2$ is satisfied; that is,

$$\Omega_{\text{bif}} = \infty \quad \text{iff} \quad W_2\Lambda_2 = \frac{(W_1\Lambda_1 + \lambda_2)^2}{4\lambda_1} \tag{4.66}$$

Define the spectral radius as $\rho(\Omega) := \max|\zeta_i|$ for $i = 1, 2$. In the low-frequency range, we have

$$\rho_0 := \lim_{\Omega\to 0} \rho(\Omega) = \lim_{\Omega\to 0} |\zeta_1| = \lim_{\Omega\to 0} |\zeta_2| = 1 \tag{4.67}$$

In the high-frequency range, the eigenvalues of $\mathbf{A}_\infty$ are given as

$$\zeta_{1,2}^{\infty} = -\frac{W_1\Lambda_1 + \lambda_2 - 2W_2\Lambda_2 \pm \sqrt{(W_1\Lambda_1 + \lambda_2)^2 - 4\lambda_1 W_2\Lambda_2}}{2W_2\Lambda_2} \tag{4.68}$$

To express W_1 and $W_2\Lambda_2$ in terms of the principal roots at the high-frequency range, set $\zeta_{1,2}^{\infty} = -\rho_{1,2\infty}$ in Eq. (4.68), and impose $\lambda_1 = \Lambda_1 = 1$ and $\lambda_2 = 1/2$, which yields

$$\boxed{W_1 = \frac{3 + \rho_\infty^{\min} + \rho_\infty^{\max} - \rho_\infty^{\min}\rho_\infty^{\max}}{2(1 + \rho_\infty^{\min})(1 + \rho_\infty^{\max})} \quad \text{and} \quad W_2\Lambda_2 = \frac{1}{(1 + \rho_\infty^{\min})(1 + \rho_\infty^{\max})}} \tag{4.69}$$

Note the spectral radius of the amplification matrix $\mathbf{A}$ is defined as $\rho_\infty := \max\{\rho_\infty^{\min}, \rho_\infty^{\max}\}$. To guarantee the second-order time accuracy of the algorithm, we must have $W_1 = 1/2$ which leads to the spectral condition $\rho_\infty^{\min}\rho_\infty^{\max} = 1$. Since the spectral condition $0 \le \rho_{i\infty} \le 1$ $(i = 1, 2)$ is required to guarantee the unconditional stability, we observe that the second-order time accurate and unconditional stable member in the framework of algorithms is obtained only for $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$, i.e., $W_1 = 1/2$ and $W_2\Lambda_2 = 1/4$, which leads to the (symplectic) midpoint rule. Notice there is no bifurcation for the midpoint rule according to Eq.(4.66). Of course, we can obtain Eq. (4.69) by equating Eq. (4.31) and

$$(\zeta + \rho_\infty^{\min})(\zeta + \rho_\infty^{\max}) = 0 \tag{4.70}$$

That is,

$$\begin{aligned} \rho_\infty^{\min} + \rho_\infty^{\max} &= -A_1^\infty \\ \rho_\infty^{\min}\rho_\infty^{\max} &= A_2^\infty \end{aligned} \tag{4.71}$$

where

$$A_1^\infty := \mathrm{tr}(\mathbf{A}_\infty) = \frac{2W_2\Lambda_2 - \lambda_2 - W_1\Lambda_1}{W_2\Lambda_2} \tag{4.72}$$

$$A_1^\infty := \det(\mathbf{A}_\infty) = \frac{\lambda_1 - W_1\Lambda_1 - \lambda_2 + W_2\Lambda_2}{W_2\Lambda_2} \tag{4.73}$$

Imposing the second-order time accurate conditions, we can obtain the spectral relation shown in Eq. (4.69).

Stability analysis by the Routh-Hurwitz conditions: Consider the characteristic polynomial of $\mathbf{A}$, i.e., Eq. (4.126). The two-field form I-GSSSS family of algorithms, which has the equivalent form of the two-stage linear multistep method, is ***absolute stable*** if the spectral radius ρ satisfies

$$\rho := \max\left(\left|\{\zeta_i\}_{i=1}^2\right|\right) \leq 1 \tag{4.74}$$

Following [29], substitute the transformation

$$\zeta = \frac{1+z}{1-z} \tag{4.75}$$

into Eq. (4.70) and multiply by $(1-z)^2$; then, we get

$$Q_0 z^2 + Q_1 z + Q_2 = 0 \tag{4.76}$$

where

$$Q_0 = 1 + A_1 + A_2 \tag{4.77}$$

$$Q_1 = 2(1 - A_2) \tag{4.78}$$

$$Q_2 = 1 - A_1 + A_2 \tag{4.79}$$

Note that the transformation, Eq. (4.31), maps the unit circle ($\rho = 1$) into the left-half $s-$plane including the imaginary axis ($\mathrm{Re}(z) \leq 0$). Involing the Routh-Hurwitz conditions [30–32], the necessary and sufficient conditions for the absolute

stability of the algorithm are therefore given by the following inequalities:

$$Q_0 \geq 0, \quad Q_1 \geq 0, \quad \text{and} \quad Q_2 \geq 0 \tag{4.80}$$

For the low-frequency limit ($\Omega \to 0$), we have

$$Q_0 = \frac{\lambda_1}{W_2 \Lambda_2} \quad \text{and} \quad Q_1 = Q_2 = 0 \tag{4.81}$$

and for the high-frequency limit ($\Omega \to \infty$), we have

$$\begin{aligned} Q_0 &= 2 + \frac{\lambda_1 - 2(\lambda_2 + W_1\Lambda_1 - W_2\Lambda_2)}{W_2\Lambda_2} \\ Q_1 &= \frac{2(W_1\Lambda_1 - \lambda_1 + \lambda_2)}{W_2\Lambda_2} \\ Q_2 &= \frac{\lambda_1}{W_2\Lambda_2} \end{aligned} \tag{4.82}$$

Imposing the second-order time accuracy conditions, we readily see that $W_2\Lambda_2 = 1/4$ yields the unconditional stability of the algorithm in both low- and high-frequency ranges (also, Eq. (4.80) is satisfied).

Measures of Accuracy: Numerical Dissipation and Numerical Dispersion

Focusing attention on the underdamped case ($0 \leq \xi < 1$); that is, $\zeta_{1,2}$ in Eq.(4.62) are complex conjugate roots which can be written as

$$\zeta_{1,2} = \exp\left[-\bar{\xi}\bar{\Omega} \pm i\bar{\Omega}_{\mathrm{d}}\right] \quad (i = \sqrt{-1}) \tag{4.83}$$

where $\bar{\Omega} = \bar{\omega}\Delta t$ and

$$\bar{\Omega}_{\mathrm{d}} = \arctan\left(\frac{\mathrm{Im}(\zeta_i)}{\mathrm{Re}(\zeta_i)}\right) = \arctan\sqrt{\frac{A_2}{A_1^2} - 1} \quad \text{with} \quad \bar{\Omega}_{\mathrm{d}} = \sqrt{1-\xi^2}\bar{\Omega} \tag{4.84}$$

$$\bar{\xi} = \frac{-1}{2\bar{\Omega}} \ln\left([\mathrm{Im}(\zeta_i)]^2 + [\mathrm{Re}(\zeta_i)]^2\right) = \frac{-1}{2\bar{\Omega}} \ln|A_2| \tag{4.85}$$

See Section 4.2 for more detailed explanations on the measures of accuracy. Setting $\lambda_1 = \Lambda_1 = \Lambda_4 = 1$ and $\lambda_2 = 1/2$ for simplicity, we obtain

$$\bar{\Omega} = \Omega + (2W_1 - 1)\,\mathcal{O}(\Omega^2) + \mathcal{O}(\Omega^3) \tag{4.86}$$

$$\bar{\xi} = \xi + (2W_1 - 1)\,\mathcal{O}(\Omega) + \mathcal{O}(\Omega^2) \tag{4.87}$$

after some laborious calculations. Therefore, we have $\bar{\Omega} = \Omega + \mathcal{O}(\Omega^2)$ and $\bar{\xi} = \xi + \mathcal{O}(\Omega)$ if $W_1 \neq 1/2$ and $\bar{\Omega} = \Omega + \mathcal{O}(\Omega^3)$ and $\bar{\xi} = \xi + \mathcal{O}(\Omega^2)$ if $W_1 = 1/2$; hence, the numerical dissipation and the numerical dispersion, i.e., the algorithmic damping ratio $\bar{\xi}$ and the relative period error defined as $P_{\text{nd}} := (\bar{T} - T)/T$ where $T := 2\pi/\omega$ and $\bar{T} := 2\pi/\bar{\omega}$, becomes first-order errors if $W_1 \neq 1/2$, which leads to first-order time accurate schemes. When $\xi = 0$ with $\lambda_1 = \Lambda_1 = \Lambda_4 = 1$, $\lambda_2 = 1/2$ and $W_1 = 1/2$, we have $P_{\text{nd}} = \mathcal{O}(\Omega^2)$ and $\bar{\xi} = 0$ for any $W_2\Lambda_2$. The numerical dissipation and dispersion are discussed in more detail in a later section.

Algorithm 1

Two-field Form I-GSSSS Framework and Family of Algorithms for Linear Dynamical Systems

Integrator:

$$\begin{aligned}&\left[\frac{1}{\Delta t}\mathbf{M} + \mathbf{C}W_1\Lambda_4 + \Delta t\mathbf{K}W_2\Lambda_2\right]\Delta\boldsymbol{\nu}\\&= -\mathbf{C}\boldsymbol{\nu}_n - \mathbf{K}\left[\mathbf{q}_n + \Delta t W_1\Lambda_1\boldsymbol{\nu}_n\right]\\&+ (1 - W_1)\mathbf{f}_n + W_1\mathbf{f}_{n+1}\end{aligned}$$

Updates:

$$\begin{aligned}\mathbf{q}_{n+1} &= \mathbf{q}_n + \Delta t\lambda_1\boldsymbol{\nu}_n + \Delta t\lambda_2\Delta\boldsymbol{\nu}\\\boldsymbol{\nu}_{n+1} &= \boldsymbol{\nu}_n + \Delta\boldsymbol{\nu}\end{aligned}$$

Initial conditions:

$$\begin{aligned}\mathbf{q}(t_0) &= \mathbf{q}_0\\\boldsymbol{\nu}(t_0) &= \boldsymbol{\nu}_0\end{aligned}$$

Remark 6 (Algorithm 1)

1. *Algorithm 1 is the single step single solve framework which can be equivalently written in the form of the linear two-step method.*

2. *There exists only one second-order time accurate and unconditionally stable scheme, and it is the midpoint rule. The spectral radii of the midpoint rule*

are $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ *for any time step size, and it is the only second-order time accurate algorithm without the bifurcation.*

Midpoint rule *(*$W_1 = \frac{1}{2}$, $W_2 = \frac{1}{2}$, $\Lambda_1 = \lambda_1 = 1$, $\Lambda_2 = \lambda_2 = \frac{1}{2}$, *and* $\Lambda_4 = 1$*):*

$$\begin{aligned} &\mathbf{M}\frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t} + \mathbf{C}\frac{\boldsymbol{\nu}_{n+1} + \boldsymbol{\nu}_n}{2} + \mathbf{K}\frac{\mathbf{q}_{n+1} + \mathbf{q}_n}{2} = \frac{\mathbf{f}_{n+1} + \mathbf{f}_n}{2} \\ &\frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t} = \frac{\boldsymbol{\nu}_{n+1} + \boldsymbol{\nu}_n}{2} \end{aligned} \tag{4.88}$$

The stability, numerical dissipation, and numerical dispersion plots are for the conservative system, $\dot{\nu} + 10q = 0$ *with* $\dot{q} = \nu$*. For* $W_1 = \frac{1}{2}$ *and* $W_2 = \frac{1}{2}$*, we have* $w_1 = -5$ *and* $w_2 = 5$*; hence, the degenerated scalar polynomial weighting time field for the midpoint rule is given as*

$$W(\tau) = 1 - 5\frac{\tau}{\Delta t} + 5\left(\frac{\tau}{\Delta t}\right)^2 \tag{4.89}$$

See Fig. 4.1 for $W(\tau)$ *when* $\Delta t = 1.0$ sec. *See Fig. 4.2 for the time accuracy, stability, numerical dissipation, and numerical dispersion plots of the algorithm (midpoint rule). The orders of accuracy in both the configuration (*$\square$*) and the velocity (*$\triangle$*) are 2 as can be seen from Fig. 4.2-(a) for* $\dot{\nu} + 0.1\nu + 10q = \sin(t)$ *with* $\dot{q} = \nu$*.*

3. *Algorithm 1 can be cast into the following form:*

$$\mathbf{M}\frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t} + \mathbf{C}\tilde{\boldsymbol{\nu}} + \mathbf{K}\tilde{\mathbf{q}} = \tilde{\mathbf{f}} \tag{4.90}$$

where

$$\begin{aligned} \tilde{\mathbf{q}} &= \left(1 - \frac{W_2\Lambda_2}{\lambda_2}\right)\mathbf{q}_n + \frac{W_2\Lambda_2}{\lambda_2}\mathbf{q}_{n+1} \\ &\quad + \Delta t\left(W_1\Lambda_1 - \frac{\lambda_1}{\lambda_2}W_2\Lambda_2\right)\boldsymbol{\nu}_n \qquad (4.91) \\ \tilde{\boldsymbol{\nu}} &= (1 - W_1\Lambda_4)\boldsymbol{\nu}_n + W_1\Lambda_4\boldsymbol{\nu}_{n+1} \qquad (4.92) \\ \tilde{\mathbf{f}} &= (1 - W_1)\mathbf{f}_n + W_1\mathbf{f}_{n+1} \qquad (4.93) \end{aligned}$$

The midpoint rule yields

$$\begin{aligned} \mathbf{0} &= \mathbf{M}\dot{\boldsymbol{\nu}}(t^*) + \mathbf{C}\boldsymbol{\nu}(t^*) + \mathbf{K}\mathbf{q}(t^*) - \mathbf{f}(t^*) \\ &= \mathbf{M}\frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t} + \mathbf{C}\boldsymbol{\nu}_{n+1/2} + \mathbf{K}\mathbf{q}_{n+1/2} - \mathbf{f}_{n+1/2} + \mathcal{O}(\Delta t^p) \end{aligned} \tag{4.94}$$

where $t^* = t_{n+\alpha}$ *for* $\alpha \in [0,1]$. *Note that* $p = 2$ *is obtained if and only if* $\alpha = 1/2$; *otherwise,* $p = 1$. *We call* $t^* = t_{n+1/2}$ *the* ***algorithmic time level*** *for the midpoint rule. It is important to note that the algorithmic time levels for* $\dot{\boldsymbol{\nu}}$, $\boldsymbol{\nu}$, $\mathbf{q}$, *and* $\mathbf{f}$ *in the equation of motion are consistent.*

4. *Algorithm 1 for linear conservative dynamical systems can be readily obtained as*

$$\begin{aligned} &\left[\frac{1}{\Delta t}\mathbf{M} + \Delta t \mathbf{K} W_2 \Lambda_2\right] \Delta\boldsymbol{\nu} = -\mathbf{K}\left[\mathbf{q}_n + \Delta t W_1 \Lambda_1 \boldsymbol{\nu}_n\right] \\ &\mathbf{q}_{n+1} = \mathbf{q}_n + \Delta t \lambda_1 \boldsymbol{\nu}_n + \Delta t \lambda_2 \Delta\boldsymbol{\nu} \\ &\boldsymbol{\nu}_{n+1} = \boldsymbol{\nu}_n + \Delta\boldsymbol{\nu} \end{aligned} \tag{4.95}$$

   by setting $\mathbf{C} = \mathbf{0}$ *and* $\mathbf{f}_n = \mathbf{f}_{n+1} = \mathbf{0}$.

5. ***Symplecticness:*** *Symplectic members within Algorithm 1 satisfy* $A_2 = \det(\mathbf{A}) = 1$ *in conservative systems, i.e.,*

$$\lambda_1 - \lambda_2 - W_1 \Lambda_1 = 0 \tag{4.96}$$

   Therefore, the mid-point rule is symplectic.

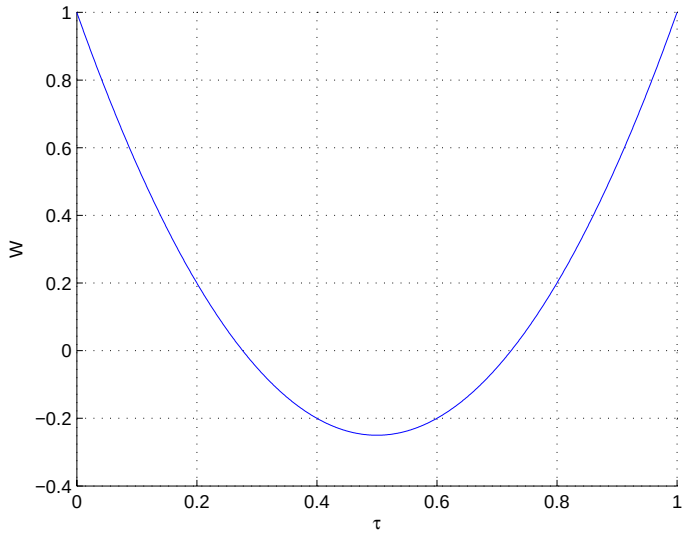
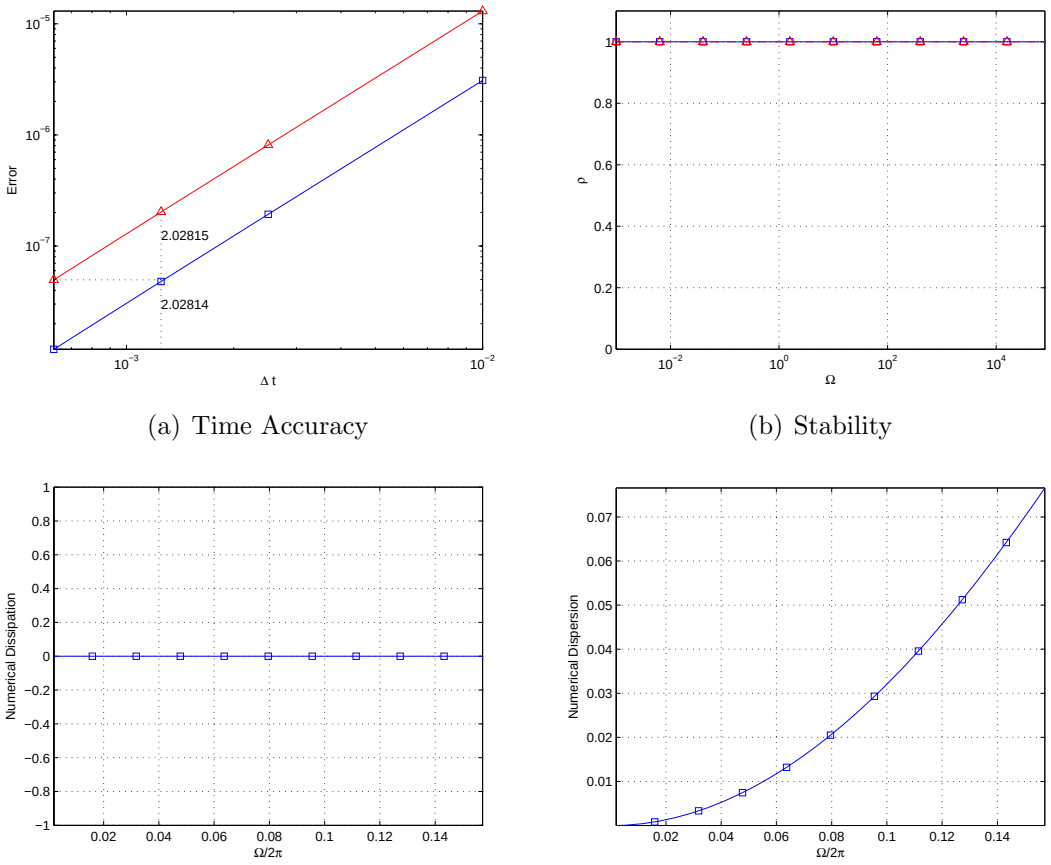


Figure 4.1: The weighting time field for the two field form midpoint rule ($\Delta t = 1.0$ sec) - Two-field form

4.2 I-GSSSS Framework and Family of Algorithms in Single-Field Form

The initial-value problem in the single-field form for linear dynamical systems consists of the linearized equation of motion and the given initial conditions. That is, in general, we have for linear dynamical systems

$$\boxed{\begin{array}{l} \textbf{\textit{Balance Equation:}} \\ \mathbf{M}\ddot{\mathbf{q}}(t) + \mathbf{N}(\mathbf{q}(t), \dot{\mathbf{q}}(t)) = \mathbf{f}(t) \quad \forall t \in \mathbb{I} \\ \text{in which } \ \mathbf{N}(\mathbf{q}(t), \dot{\mathbf{q}}(t)) = \mathbf{C}\dot{\mathbf{q}}(t) + \mathbf{K}\mathbf{q}(t) \\ \textbf{\textit{Initial conditions:}} \\ \mathbf{q}(t_0) = \mathbf{q}_0 \\ \dot{\mathbf{q}}(t_0) = \dot{\mathbf{q}}_0 \end{array}} \tag{4.97}$$



(a) Time Accuracy

(b) Stability

(c) Numerical Dissipation

(d) Numerical Dispersion

Figure 4.2: Midpoint rule plots of time accuracy, stability, numerical dissipation, and numerical dispersion of Algorithm 1 - Two-field form

From Eq. (4.97), we have

$$\mathbf{M}\ddot{\mathbf{q}}(t) + \mathbf{C}\dot{\mathbf{q}}(t) + \mathbf{K}\mathbf{q}(t) - \mathbf{f}(t) = \mathbf{0} \quad \forall t \in \mathbb{I} \tag{4.98}$$

The asymptotic series expansion of the dependent variables, $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\ddot{\mathbf{q}}$, about time $t = t_{n+\alpha} := (1-\alpha)t_n + \alpha t_{n+1} \in [t_n, t_{n+1}]$ for $\alpha \in [0,1]$ yield

$$\begin{aligned}
\mathbf{q}(t_{n+\alpha}) &\cong \mathbf{q}(t_n) + \Lambda_1 \dot{\mathbf{q}}(t_n)[t_{n+\alpha} - t_n] + \Lambda_2 \ddot{\mathbf{q}}(t_n)[t_{n+\alpha} - t_n]^2 \\
&\quad + \Lambda_3 \frac{\ddot{\mathbf{q}}(t_{n+1}) - \ddot{\mathbf{q}}(t_n)}{\Delta t}[t_{n+\alpha} - t_n]^3 \qquad (4.99) \\
\dot{\mathbf{q}}(t_{n+\alpha}) &\cong \dot{\mathbf{q}}(t_n) + \Lambda_4 \ddot{\mathbf{q}}(t_n)[t_{n+\alpha} - t_n] \\
&\quad + \Lambda_5 \frac{\ddot{\mathbf{q}}(t_{n+1}) - \ddot{\mathbf{q}}(t_n)}{\Delta t}[t_{n+\alpha} - t_n]^2 \qquad (4.100) \\
\ddot{\mathbf{q}}(t_{n+\alpha}) &\cong \ddot{\mathbf{q}}(t_n) + \Lambda_6 \frac{\ddot{\mathbf{q}}(t_{n+1}) - \ddot{\mathbf{q}}(t_n)}{\Delta t}[t_{n+\alpha} - t_n] \qquad (4.101)
\end{aligned}$$

respectively with algorithmic parameters $\Lambda_i \in \mathbb{R}$ $(i = 1,2,3,4,5,6)$. For the time-dependent external force vector, we linearly approximate within a time step $[t_n, t_{n+1}]$ as

$$\mathbf{f}(t_{n+\alpha}) \cong \mathbf{f}(t_n) + \frac{\mathbf{f}(t_{n+1}) - \mathbf{f}(t_n)}{\Delta t}[t_{n+\alpha} - t_n] \qquad (4.102)$$

Defining $\tau := t_{n+\alpha} - t_n \in [0, \Delta t]$, we have [7]

$$\begin{aligned}
\hat{\mathbf{q}}(\tau) &:= \mathbf{q}_n + \Lambda_1 \dot{\mathbf{q}}_n \tau + \Lambda_2 \ddot{\mathbf{q}}_n \tau^2 + \Lambda_3 \frac{\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n}{\Delta t}\tau^3 \qquad (4.103) \\
\hat{\mathbf{v}}(\tau) &:= \dot{\mathbf{q}}_n + \Lambda_4 \ddot{\mathbf{q}}_n \tau + \Lambda_5 \frac{\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n}{\Delta t}\tau^2 \qquad (4.104) \\
\hat{\mathbf{a}}(\tau) &:= \ddot{\mathbf{q}}_n + \Lambda_6 \frac{\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n}{\Delta t}\tau \qquad (4.105) \\
\hat{\mathbf{f}}(\tau) &:= \mathbf{f}_n + \frac{\mathbf{f}_{n+1} - \mathbf{f}_n}{\Delta t}\tau \qquad (4.106)
\end{aligned}$$

Substituting $\mathbf{q} \approx \hat{\mathbf{q}}$, $\dot{\mathbf{q}} \approx \hat{\mathbf{v}}$, $\ddot{\mathbf{q}} \approx \hat{\mathbf{a}}$, and $\mathbf{f} \approx \hat{\mathbf{f}}$ into (4.98), the residual is defined as

$$\mathbf{r} := \mathbf{M}\hat{\mathbf{a}} + \mathbf{C}\hat{\mathbf{v}} + \mathbf{K}\hat{\mathbf{q}} - \hat{\mathbf{f}} \qquad (4.107)$$

Note that the residual is non-zero unless otherwise $\hat{\mathbf{a}}$, $\hat{\mathbf{v}}$, $\hat{\mathbf{q}}$, and $\hat{\mathbf{f}}$ are exact. Employing the ***generalized time weighted residual methodology*** within the time step $[t_n, t_{n+1}]$ for the residual above, $\mathbf{r}$, we have

$$\frac{\int_0^{\Delta t} W \mathbf{r} d\tau}{\int_0^{\Delta t} W d\tau} = \frac{\int_0^{\Delta t} W \left[\mathbf{M}\hat{\mathbf{a}} + \mathbf{C}\hat{\mathbf{v}} + \mathbf{K}\hat{\mathbf{q}} - \hat{\mathbf{f}}\right] d\tau}{\int_0^{\Delta t} W d\tau} \cong \mathbf{0} \qquad (4.108)$$

or

$$\tilde{\mathbf{r}} = \mathbf{M}\tilde{\mathbf{a}} + \mathbf{C}\tilde{\mathbf{v}} + \mathbf{K}\tilde{\mathbf{q}} - \tilde{\mathbf{f}} \cong \mathbf{0} \tag{4.109}$$

where the algorithmic unknowns are defined by Eq. (4.12). A degenerated scalar polynomial weighting time field, $W \in \mathbb{R}$, in single-field form is defined as

$$W(\tau) \cong w_0 + w_1 \frac{\tau}{\Delta t} + w_2 \left(\frac{\tau}{\Delta t}\right)^2 + w_3 \left(\frac{\tau}{\Delta t}\right)^3 \tag{4.110}$$

with $w_i \in \mathbb{R}$ (for $i = 0, 1, 2, 3$). Again, set $w_0 = 1$ ($w_0 \neq 0$ in general) [28]. Replacing

$$\begin{aligned}
\frac{\int_0^{\Delta t} \frac{\tau}{\Delta t} W d\tau}{\int_0^{\Delta t} W d\tau} &= \frac{30 + 20w_1 + 15w_2 + 12w_3}{60 + 30w_1 + 20w_2 + 15w_3} \\
\frac{\int_0^{\Delta t} \left(\frac{\tau}{\Delta t}\right)^2 W d\tau}{\int_0^{\Delta t} W d\tau} &= \frac{20 + 15w_1 + 12w_2 + 10w_3}{60 + 30w_1 + 20w_2 + 15w_3} \\
\frac{\int_0^{\Delta t} \left(\frac{\tau}{\Delta t}\right)^3 W d\tau}{\int_0^{\Delta t} W d\tau} &= \frac{105 + 84w_1 + 70w_2 + 60w_3}{420 + 210w_1 + 140w_2 + 105w_3}
\end{aligned} \tag{4.111}$$

with algorithmic parameters, W_1, W_2, and W_3, respectively, the algorithmic unknowns are given as

$$\tilde{\mathbf{q}} = \mathbf{q}_n + W_1\Lambda_1\dot{\mathbf{q}}_n\Delta t + W_2\Lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + W_3\Lambda_3\left(\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n\right)\Delta t^2 \tag{4.112}$$

$$\tilde{\mathbf{v}} = \dot{\mathbf{q}}_n + W_1\Lambda_4\ddot{\mathbf{q}}_n\Delta t + W_2\Lambda_5(\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n)\Delta t \tag{4.113}$$

$$\tilde{\mathbf{a}} = \ddot{\mathbf{q}}_n + W_1\Lambda_6(\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n) \tag{4.114}$$

$$\tilde{\mathbf{f}} = \mathbf{f}_n + W_1(\mathbf{f}_{n+1} - \mathbf{f}_n) \tag{4.115}$$

The corresponding generalized updates are designed by setting $\alpha = 1$ in Eq. (4.99) and Eq. (4.100) and replacing Λ_i with new algorithmic parameters $\lambda_i \in \mathbb{R}$ ($i = 1, 2, 3, 4, 5$), i.e.,

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + \lambda_3(\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n)\Delta t^2 \tag{4.116}$$

$$\dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5(\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n)\Delta t \tag{4.117}$$

Rearranging the representation of the algorithm given above leads to the general form of the implicit GSSSS framework and family of algorithms in single-field

form (or the ***single-field form I-GSSSS framework***) as

A-form

$$
\begin{aligned}
&\left[W_1\Lambda_6\mathbf{M} + W_2\Lambda_5\mathbf{C}\Delta t + W_3\Lambda_3\mathbf{K}\Delta t^2\right]\Delta\mathbf{a} \\
&= -\mathbf{M}\ddot{\mathbf{q}}_n - \mathbf{C}\left[\dot{\mathbf{q}}_n + W_1\Lambda_4\ddot{\mathbf{q}}_n\Delta t\right] \\
&- \mathbf{K}\left[\mathbf{q}_n + W_1\Lambda_1\dot{\mathbf{q}}_n\Delta t + W_2\Lambda_2\ddot{\mathbf{q}}_n\Delta t^2\right] \\
&+ (1 - W_1)\mathbf{f}_n + W_1\mathbf{f}_{n+1}
\end{aligned}
$$

Updates:

$$
\begin{aligned}
\mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + \lambda_3\Delta\mathbf{a}\Delta t^2 \\
\dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5\Delta\mathbf{a}\Delta t \\
\ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_n + \Delta\mathbf{a}
\end{aligned}
$$

Initial conditions:

$$
\begin{aligned}
\mathbf{q}(t_0) &= \mathbf{q}_0 \\
\dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0
\end{aligned}
$$

(4.118)

where the $\mathbf{a}$-increment $\Delta\mathbf{a} := \ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n$ is the primary unknown in the ***A-form*** representation of the single-field form I-GSSSS framework. Equivalently, the ***V-form*** and ***D-form*** representations are readily obtained and they all yield identical results.

V-form

$$\left[\frac{W_1\Lambda_6}{\lambda_5\Delta t}\mathbf{M} + \frac{W_2\Lambda_5}{\lambda_5}\mathbf{C} + \frac{W_3\Lambda_3}{\lambda_5}\mathbf{K}\Delta t\right]\Delta\mathbf{v}$$
$$= -\left[1 - W_1\Lambda_6\frac{\lambda_4}{\lambda_5}\right]\mathbf{M}\ddot{\mathbf{q}}_n - \mathbf{C}\left[\dot{\mathbf{q}}_n + \left(W_1\Lambda_4 - W_2\Lambda_5\frac{\lambda_4}{\lambda_5}\right)\ddot{\mathbf{q}}_n\Delta t\right]$$
$$- \mathbf{K}\left[\mathbf{q}_n + W_1\Lambda_1\dot{\mathbf{q}}_n\Delta t + \left(W_2\Lambda_2 - W_3\Lambda_3\frac{\lambda_4}{\lambda_5}\right)\ddot{\mathbf{q}}_n\Delta t^2\right]$$
$$+ (1 - W_1)\mathbf{f}_n + W_1\mathbf{f}_{n+1}$$

Updates:

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \left(\lambda_2 - \frac{\lambda_3\lambda_4}{\lambda_5}\right) + \frac{\lambda_3}{\lambda_5}\Delta\mathbf{v}\Delta t$$
$$\ddot{\mathbf{q}}_{n+1} = \left(1 - \frac{\lambda_4}{\lambda_5}\right)\ddot{\mathbf{q}}_n + \frac{1}{\lambda_5\Delta t}\Delta\mathbf{v}$$
$$\dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_n + \Delta\mathbf{v}$$

Initial conditions:

$$\mathbf{q}(t_0) = \mathbf{q}_0$$
$$\dot{\mathbf{q}}(t_0) = \dot{\mathbf{q}}_0$$

(4.119)

D-form

$$\left[\frac{W_1\Lambda_6}{\lambda_3\Delta t^2}\mathbf{M}+\frac{W_2\Lambda_5}{\lambda_3\Delta t}\mathbf{C}+\frac{W_3\Lambda_3}{\lambda_3}\mathbf{K}\Delta t\right]\Delta\mathbf{q}$$
$$=\mathbf{M}\left[W_1\Lambda_6\frac{\lambda_1}{\lambda_3\Delta t}\dot{\mathbf{q}}_n-\left(1-W_1\Lambda_6\frac{\lambda_2}{\lambda_3}\right)\ddot{\mathbf{q}}_n\right]$$
$$-\mathbf{C}\left[\left(1-W_2\Lambda_5\frac{\lambda_1}{\lambda_3}\right)\dot{\mathbf{q}}_n+\left(W_1\Lambda_4-W_2\Lambda_5\frac{\lambda_2}{\lambda_3}\right)\ddot{\mathbf{q}}_n\Delta t\right]$$
$$-\mathbf{K}\left[\mathbf{q}_n+\left(W_1\Lambda_1-W_3\Lambda_3\frac{\lambda_1}{\lambda_3}\right)\dot{\mathbf{q}}_n\Delta t+\left(W_2\Lambda_2-W_3\Lambda_3\frac{\lambda_2}{\lambda_3}\right)\dot{\mathbf{q}}_n\Delta t\right]$$
$$+(1-W_1)\mathbf{f}_n+W_1\mathbf{f}_{n+1} \tag{4.120}$$

Updates:

$$\mathbf{q}_{n+1}=\left(1-\frac{\lambda_1\lambda_5}{\lambda_3}\right)\dot{\mathbf{q}}_n+\left(\lambda_4-\frac{\lambda_2\lambda_5}{\lambda_3}\right)\ddot{\mathbf{q}}_n\Delta t+\frac{\lambda_5}{\lambda_3\Delta t}\Delta\mathbf{q}$$
$$\ddot{\mathbf{q}}_{n+1}=-\frac{\lambda_1}{\lambda_3\Delta t}+\left(1-\frac{\lambda_2}{\lambda_3}\right)\ddot{\mathbf{q}}_n+\frac{1}{\lambda_3\Delta t^2}\Delta\mathbf{q}$$
$$\mathbf{q}_{n+1}=\mathbf{q}_n+\Delta\mathbf{q}$$

Initial conditions:

$$\mathbf{q}(t_0)=\mathbf{q}_0$$
$$\dot{\mathbf{q}}(t_0)=\dot{\mathbf{q}}_0$$

The algorithm given in Eq. (4.118) (which is in A-form) can be transformed

into the SDOF equation via the modal decomposition as

$$\begin{aligned}
&\left[W_1\Lambda_6 + 2W_2\Lambda_5\xi\omega\Delta t + W_3\Lambda_3\omega^2\Delta t^2\right]\Delta a \\
&= -\ddot{q}_n - 2\xi\omega\left[\dot{q}_n + W_1\Lambda_4\ddot{q}_n\Delta t\right] - \omega^2\left[q_n + W_1\Lambda_1\dot{q}_n\Delta t + W_2\Lambda_2\ddot{q}_n\Delta t^2\right] \\
&+ (1 - W_1)g_n + W_1 g_{n+1}
\end{aligned}$$

Updates:

$$\begin{aligned}
q_{n+1} &= q_n + \lambda_1\dot{q}_n\Delta t + \lambda_2\ddot{q}_n\Delta t^2 + \lambda_3\Delta a\Delta t^2 \\
\dot{q}_{n+1} &= \dot{q}_n + \lambda_4\ddot{q}_n\Delta t + \lambda_5\Delta a\Delta t \\
\ddot{q}_{n+1} &= \ddot{q}_n + \Delta a
\end{aligned} \tag{4.121}$$

Initial conditions:

$$\begin{aligned}
q(t_0) &= q_0 \\
\dot{q}(t_0) &= \dot{q}_0
\end{aligned}$$

Defining $\mathbf{y}_{n+1} := \{q_{n+1}, \Delta t\dot{q}_{n+1}, \Delta t^2\ddot{q}_{n+1}\}^T$ and $\mathbf{y}_n := \{q_n, \Delta t\dot{q}_n, \Delta t^2\ddot{q}_n\}^T$, Eq. (4.121) can be written as

$$\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n + \mathbf{L}_{n+W_1} \tag{4.122}$$

where the amplification matrix $\mathbf{A}$ for the single-field form I-GSSSS family of algorithms is given as

$$\mathbf{A} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} = \begin{bmatrix} 1 & \lambda_1 & \lambda_2 \\ 0 & 1 & \lambda_4 \\ 0 & 0 & 1 \end{bmatrix} + \begin{pmatrix} \lambda_3 \\ \lambda_5 \\ 1 \end{pmatrix} \begin{pmatrix} \alpha_1 & \alpha_2 & \alpha_3 \end{pmatrix} \tag{4.123}$$

in which

$$\begin{aligned}
\alpha_1 &= -\frac{\Omega^2}{D} \\
\alpha_2 &= -\frac{2\xi\Omega + W_1\Lambda_1\Omega^2}{D} \\
\alpha_3 &= -\frac{1 + 2W_1\Lambda_4\xi\Omega + W_2\Lambda_2\Omega^2}{D} \\
D &= W_1\Lambda_6 + 2W_2\Lambda_5\xi\Omega + W_3\Lambda_3\Omega^2 \\
\Omega &= \omega\Delta t
\end{aligned} \tag{4.124}$$

and the external load vector is given as

$$\mathbf{L}_{n+W_1} = \frac{\Delta t^2}{D}[(1-W_1)g_n + W_1 g_{n+1}] \begin{pmatrix} \lambda_3 \\ \lambda_5 \\ 1 \end{pmatrix} \tag{4.125}$$

The characteristic polynomial of the amplification matrix is given as

$$-\det(\zeta \mathbf{I}_3 - \mathbf{A}) = \zeta^3 - A_1\zeta^2 + A_2\zeta - A_3 = 0 \tag{4.126}$$

where ζ are the eigenvalues of $\mathbf{A}$; $\mathbf{I}_3$ denotes the 3×3 identity matrix; and A_i $(i = 1, 2, 3)$ are defined as

$$A_1 = \operatorname{tr}(\mathbf{A}) = \zeta_1 + \zeta_2 + \zeta_3 \tag{4.127}$$

$$A_2 = \frac{1}{2}\left[(\operatorname{tr}(\mathbf{A}))^2 - \operatorname{tr}(\mathbf{A}^2)\right] = \zeta_1\zeta_2 + \zeta_2\zeta_3 + \zeta_3\zeta_1 \tag{4.128}$$

$$A_3 = \det(\mathbf{A}) = \zeta_1\zeta_2\zeta_3 \tag{4.129}$$

Employing the Cayley-Hamilton theorem, we get

$$\mathbf{A}^3 - A_1\mathbf{A}^2 + A_2\mathbf{A} - A_3\mathbf{I}_3 = \mathbf{0} \tag{4.130}$$

For the homogeneous case, i.e., $\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n$,

$$\begin{aligned} \mathbf{y}_{n+2} &= \mathbf{A}^3\mathbf{y}_{n-1} \\ &= \left(A_1\mathbf{A}^2 - A_2\mathbf{A} + A_3\mathbf{I}_3\right)\mathbf{y}_{n-1} \\ &= A_1\mathbf{y}_{n+1} - A_2\mathbf{y}_n + A_3\mathbf{y}_{n-1} \end{aligned} \tag{4.131}$$

Therefore, the difference equation is obtained as

$$\mathbf{y}_{n+2} - A_1\mathbf{y}_{n+1} + A_2\mathbf{y}_n - A_3\mathbf{y}_{n-1} = \mathbf{0} \tag{4.132}$$

For the non-homogeneous case, i.e., $\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n + \mathbf{L}_{n+W_1}$, the difference equation yields

$$\begin{aligned} &\mathbf{y}_{n+2} - A_1\mathbf{y}_{n+1} + A_2\mathbf{y}_n - A_3\mathbf{y}_{n-1} \\ &= \left(\mathbf{A}^2 - A_1\mathbf{A} + A_2\mathbf{I}_3\right)\mathbf{L}_{n-1+W_1} + \left(\mathbf{A} - A_1\mathbf{I}\right)\mathbf{L}_{n+W_1} + \mathbf{L}_{n+1+W_1} \end{aligned} \tag{4.133}$$

Order of Time Accuracy

Consider the homogeneous case for simplicity. Define the local truncation error of the linear multi-step (LMS) representation for q from the difference equation given by Eq. (4.132) as

$$\tau_q^{(1)}(\Delta t) := \frac{1}{\Delta t^2}\left[q(t_{n+2}) - A_1 q(t_{n+1}) + A_2 q(t_n) - A_3 q(t_{n-1})\right] \tag{4.134}$$

where $q(t_{n+2})$, $q(t_{n+1})$, $q(t_n)$, and $q(t_{n-1})$ denote the exact solutions at time t_{n+2}, t_{n+1}, t_n, and t_{n-1}, respectively. Substituting the Taylor series expansions of $q(t_{n+2})$, $q(t_{n+1})$, and $q(t_{n-1})$ about time t_n, i.e.,

$$q(t_{n+2}) = q(t_n) + 2\Delta t\dot{q}(t_n) + 2\Delta t^2\ddot{q}(t_n) + \mathcal{O}(\Delta t^3) \tag{4.135}$$

$$q(t_{n+1}) = q(t_n) + \Delta t\dot{q}(t_n) + \frac{\Delta t^2}{2}\ddot{q}(t_n) + \mathcal{O}(\Delta t^3) \tag{4.136}$$

$$q(t_{n-1}) = q(t_n) - \Delta t\dot{q}(t_n) + \frac{\Delta t^2}{2}\ddot{q}(t_n) + \mathcal{O}(\Delta t^3) \tag{4.137}$$

we get

$$\begin{aligned}\tau_q^{(1)}(\Delta t) = \frac{1}{D}&\left[\ddot{q}(t_n) + 2\lambda_4\xi\omega\dot{q}(t_n) + \lambda_1\lambda_4\omega^2 q(t_n)\right] \\ &+ \frac{\Delta t}{D}[W_1\Lambda_6\dddot{q}(t_n) + 2\left(\lambda_5 + W_1\Lambda_4 - \frac{\lambda_4}{2}\right)\xi\omega\ddot{q}(t_n) \\ &+ \left(\lambda_2 - \lambda_1\lambda_4 + \lambda_1\lambda_5 + \lambda_4 W_1\Lambda_1\right)\omega^2\dot{q}(t_n)] + \mathcal{O}(\Delta t^2)\end{aligned} \tag{4.138}$$

Hence, for $\tau_q^{(1)}(\Delta t) = \mathcal{O}(\Delta t)$, we must have $\lambda_1 = \lambda_4 = 1$ and all other algorithmic parameters are arbitrary; and for $\tau_q^{(1)}(\Delta t) = \mathcal{O}(\Delta t^2)$, we must have

$$\begin{aligned}&\lambda_1 = \lambda_4 = 1 \\ &W_1\Lambda_6 = \lambda_5 + W_1\Lambda_4 - \frac{\lambda_4}{2} = \lambda_2 - \lambda_1\lambda_4 + \lambda_1\lambda_5 + \lambda_4 W_1\Lambda_1\end{aligned} \tag{4.139}$$

and all other algorithmic parameters are arbitrary. Next, define the local truncation error for q from $\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n$ as

$$\tau_q^{(2)}(\Delta t) := q(t_{n+1}) - A_{11}q(t_n) - \Delta t A_{12}\dot{q}(t_n) - \Delta t^2 A_{13}\ddot{q}(t_n) \tag{4.140}$$

Substituting the Taylor series expansion of $q(t_{n+1})$ about time t_n, we get

$$\begin{aligned}\tau_q^{(2)}(\Delta t) := & \frac{\Delta t}{D} W_1 \Lambda_6 (1-\lambda_1)\dot{q}(t_n) \\ & + \frac{\Delta t^2}{2D}[\left(\lambda_3 + W_1\Lambda_6(\frac{1}{2}-\lambda_2)\right)\ddot{q}(t_n) \\ & + 2\left(\lambda_3 - W_2\Lambda_5(\lambda_1-1)\right)\xi\omega\dot{q}(t_n) + \lambda_3\omega^2 q(t_n)] + \mathcal{O}(\Delta t^3)\end{aligned} \tag{4.141}$$

Hence, $\tau_q^{(2)}(\Delta t) = \mathcal{O}(\Delta t^2)$ if $\lambda_1 = 1$ and all other algorithmic parameters are arbitrary; and $\tau_q^{(2)}(\Delta t) = \mathcal{O}(\Delta t^3)$ if

$$\lambda_1 = 1 \ \text{ and } \ \lambda_2 = \frac{1}{2} \tag{4.142}$$

and all other algorithmic parameters are arbitrary. Hence, from the requirements for both $\tau_q^{(1)}(\Delta t) = \mathcal{O}(\Delta t^2)$ and $\tau_q^{(2)}(\Delta t) = \mathcal{O}(\Delta t^3)$, the necessary and sufficient conditions of the second-order time accuracy for homogeneous case are given as follows:

$$\boxed{\begin{aligned} \lambda_1 &= \lambda_4 = 1 \\ \lambda_2 &= \frac{1}{2} \\ \lambda_5 &= \frac{1}{2}\left[1 + 2W_1(\Lambda_6 - \Lambda_1)\right] \\ \Lambda_1 &= \Lambda_4 \\ \text{All other parameters}: &\quad \text{arbitrary} \\ &\textbf{(for homogeneous case)} \end{aligned}} \tag{4.143}$$

The necessary and sufficient conditions of the second-order time accuracy for non-homogeneous case are:

$$\boxed{\begin{aligned} \lambda_1 &= \lambda_4 = 1 \\ \lambda_2 &= \frac{1}{2} \\ \lambda_5 &= \frac{1}{2}\left[1 + 2W_1(\Lambda_6 - 1)\right] \\ \Lambda_1 &= \Lambda_4 = 1 \\ \text{All other parameters}: &\quad \text{arbitrary} \\ &\textbf{(for non-homogeneous case)} \end{aligned}} \tag{4.144}$$

The procedure is straightforward; therefore, the detailed analysis is omitted.

Stability

Consider the characteristic polynomial of $\mathbf{A}$, i.e., Eq. (4.126). The single-field form I-GSSSS family of algorithms is ***absolute stable*** if the spectral radius satisfies

$$\rho := \max\left(\left|\{\zeta_i\}_{i=1}^{3}\right|\right) \leq 1 \tag{4.145}$$

Following [29], substitute the transformation

$$\phi = \frac{1+z}{1-z} \tag{4.146}$$

into Eq. (4.126) and multiply by $(1-z)^3$; then, we get

$$Q_0 z^3 + Q_1 z^2 + Q_2 z + Q_3 = 0 \tag{4.147}$$

where

$$Q_0 = 1 - A_1 + A_2 - A_3 \tag{4.148}$$

$$Q_1 = 3 - A_1 - A_2 + 3A_3 \tag{4.149}$$

$$Q_2 = 3 + A_1 - A_2 - 3A_3 \tag{4.150}$$

$$Q_3 = 1 + A_1 + A_2 + A_3 \tag{4.151}$$

Note that the transformation, Eq. (4.146), maps the unit circle ($\rho = 1$) into the left-half $s-$plane including the imaginary axis ($\mathrm{Re}(z) \leq 0$). Involing the Routh-Hurwitz conditions [30–32], the necessary and sufficient conditions for the absolute stability of the algorithm are therefore given by the following inequalities:

$$\begin{aligned} &Q_0 \geq 0, \;\; Q_1 \geq 0 \\ &Q_2 \geq 0, \;\; Q_3 \geq 0 \\ &Q_4 = \frac{1}{8}\left(Q_1 Q_2 - Q_0 Q_3\right) \\ &\quad = 1 - A_2 + A_3(A_1 - A_3) \geq 0 \end{aligned} \tag{4.152}$$

The amplification matrix $\mathbf{A}$ at the low frequency limit yields

$$\mathbf{A}_0 = \lim_{\Omega \to 0} \mathbf{A} = \begin{pmatrix} 1 & \lambda_1 & \lambda_2 - \frac{\lambda_3}{W_1 \Lambda_6} \\ 0 & 1 & \lambda_4 - \frac{\lambda_5}{W_1 \Lambda_6} \\ 0 & 0 & 1 - \frac{1}{W_1 \Lambda_6} \end{pmatrix} \tag{4.153}$$

The eigenvalues of $\mathbf{A}_0$ are given as

$$\zeta_{1,2}^0 = 1, \;\; \text{and} \;\; \zeta_3^0 = 1 - \frac{1}{W_1 \Lambda_6} \tag{4.154}$$

Therefore, the spectral radius and the spurious root at the low frequency are

$$\rho^0 = \rho_{1,2}^0 = 1 \;\; \text{and} \;\; \rho_3^0 = \left| 1 - \frac{1}{W_1 \Lambda_6} \right| \tag{4.155}$$

Overshoot

Consider the SDOF homogeneous equation, $\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n$. When an unconditionally stable algorithm is applied to multi-degree of freedom problems, we tend to have the so-called high-frequency phenomenon, i.e., large values of Ω can occur. In order to investigate the overshooting behavior of the algorithm, we approximate the first time step of the scheme with $\mathbf{A}_\infty := \lim_{\Omega \to \infty} \mathbf{A}$. That is,

$$\mathbf{y}_1 \simeq \mathbf{A}_\infty \mathbf{y}_0 \tag{4.156}$$

where

$$\mathbf{A}_\infty = \begin{bmatrix} 1-\frac{\lambda_3}{W_3\Lambda_3} & \lambda_1-\frac{\lambda_3 W_1\Lambda_1}{W_3\Lambda_3} & \lambda_2-\frac{\lambda_3 W_2\Lambda_2}{W_3\Lambda_3} \\ -\frac{\lambda_5}{W_3\Lambda_3} & 1-\frac{\lambda_5 W_1\Lambda_1}{W_3\Lambda_3} & \lambda_4-\frac{\lambda_5 W_2\Lambda_2}{W_3\Lambda_3} \\ -\frac{1}{W_3\Lambda_3} & -\frac{W_1\Lambda_1}{W_3\Lambda_3} & 1-\frac{W_2\Lambda_2}{W_3\Lambda_3} \end{bmatrix} \tag{4.157}$$

and

$$\mathbf{y}_0 = \begin{pmatrix} q_0 \\ \Delta t\dot{q}_0 \\ \Delta t^2\ddot{q}_0 \end{pmatrix} = \begin{pmatrix} q_0 \\ \Delta t\dot{q}_0 \\ -2\xi\Omega\Delta t\dot{q}_0 - \Omega^2 q_0 \end{pmatrix} \tag{4.158}$$

Imposing the necessary and sufficient second-order time accuracy conditions, Eq. (4.143), yields the displacement and velocity overshoot properties as

$$\begin{aligned} q_1 \simeq & \left[1-\frac{\lambda_3}{W_3\Lambda_3}+\Omega^2\left(\frac{\lambda_3 W_2\Lambda_2}{W_3\Lambda_3}-\frac{1}{2}\right)\right] q_0 \\ & \left[1-\frac{\lambda_3 W_1\Lambda_1}{W_3\Lambda_3}+2\xi\Omega\left(\frac{\lambda_3 W_2\Lambda_2}{W_3\Lambda_3}-\frac{1}{2}\right)\right]\dot{q}_0\Delta t \end{aligned} \tag{4.159}$$

and

$$\begin{aligned} \dot{q}_1 \simeq & \frac{1}{W_3\Lambda_3}\left[W_1\Lambda_4 - W_1\Lambda_6 - \frac{1}{2}\right. \\ & \left.+\Omega^2\left(\frac{W_2\Lambda_2}{2}+W_2\Lambda_2(W_1\Lambda_6-W_1\Lambda_4)-W_3\Lambda_3\right)\right]\frac{q_0}{\Delta t} \\ & +\frac{1}{W_3\Lambda_3}\left[W_3\Lambda_3+W_1\Lambda_1\left(W_1\Lambda_4-W_1\Lambda_6\right)-\frac{W_1\Lambda_1}{2}\right. \\ & \left.+2\xi\Omega\left(\frac{W_2\Lambda_2}{2}+W_2\Lambda_2(W_1\Lambda_6-W_1\Lambda_4)-W_3\Lambda_3\right)\right]\dot{q}_0 \end{aligned} \tag{4.160}$$

respectively. Hence, the zero-order displacement overshoot (U0) constraints are given as

$$\frac{\lambda_3 W_2\Lambda_2}{W_3\Lambda_3} = \frac{1}{2} \text{ and } \frac{\lambda_3 W_1\Lambda_1}{W_3\Lambda_3} = 1 \tag{4.161}$$

or

$$\boxed{W_1\Lambda_1 = 2W_2\Lambda_2 = \frac{W_3\Lambda_3}{\lambda_3}} \tag{4.162}$$

so that Eq. (4.159) becomes

$$q_1 \simeq \left[1-\frac{\lambda_3}{W_3\Lambda_3}\right] q_0 \tag{4.163}$$

Similarly, the zero-order velocity overshoot (V0) constraint is given as

$$\boxed{\frac{W_2\Lambda_2}{2} + W_2\Lambda_2(W_1\Lambda_6 - W_1\Lambda_4) - W_3\Lambda_3 = 0} \tag{4.164}$$

so that Eq. (4.160) becomes

$$\begin{aligned}\dot{q}_1 \simeq & \frac{1}{W_3\Lambda_3}\left[W_1(\Lambda_4 - \Lambda_6) - \frac{1}{2}\right]\frac{q_0}{\Delta t} \\ & + \frac{1}{W_3\Lambda_3}\left[W_3\Lambda_3 + W_1^2\Lambda_1\left(\Lambda_4 - \Lambda_6\right) - \frac{W_1\Lambda_1}{2}\right]\dot{q}_0\end{aligned} \tag{4.165}$$

The general class of linear three-step methods developed over the past fifty years or so and new developments to-date can be characterized by two distinct families of algorithmic structures, namely, constrained U (configuration overshooting aspects) and constrained V (velocity overshooting aspects). Only those with either zero-order configuration overshooting algorithms (*U0 family of algorithms*) or zero-order velocity overshooting algorithms (*V0 family of algorithms*) are competitive amongst the class of all linear three-step methods. As a consequence, all these others are omitted here. The U0 and V0 represent zero-order overshooting behaviors in the configuration and the velocity fields, respectively.

U0-Family of I-GSSSS Algorithms by Design

Next, we design the family of algorithms with the zero-order displacement overshoot within the I-GSSSS family of algorithms (***U0-Family*** of I-GSSSS algorithms) in terms of the principal roots and spurious root at the high-frequency limit, i.e.,

$$\rho_{i\infty} := \lim_{\Omega\to\infty} \rho_i(\mathbf{A}) \tag{4.166}$$

for $i = 1, 2, 3$. At the high-frequency limit, $\mathbf{A}$ has three distinct real roots. Define the maximum principal root as $\rho_\infty^{\max} = \max(\rho_{1\infty}, \rho_{2\infty}, \rho_{3\infty})$ and the spurious root as $\rho_\infty^{\mathrm{s}} = \min(\rho_{1\infty}, \rho_{2\infty}, \rho_{3\infty})$. The three roots satisfy

$$0 \le \rho_\infty^{\mathrm{s}} \le \rho_\infty^{\min} \le \rho_\infty^{\max} \le 1 \tag{4.167}$$

where $\rho_\infty^{\min}$ is called the minimum principal root. Therefore, the characteristic polynomial can be written as

$$(\zeta + \rho_\infty^{\min})(\zeta + \rho_\infty^{\max})(\zeta + \rho_\infty^{\mathrm{s}}) = 0 \tag{4.168}$$

At the high-frequency limit, the characteristic polynomial of $\mathbf{A}$, i.e., Eq. (4.126), yields

$$\zeta^3 - A_1^\infty \zeta^2 + A_2^\infty \zeta - A_3^\infty = 0 \tag{4.169}$$

where

$$A_1^\infty = \frac{6W_2\Lambda_2 W_3\Lambda_3 - W_3\Lambda_3 - 4(W_2\Lambda_2)^2(1 - W_1\Lambda_4 + W_1\Lambda_6)}{2W_2\Lambda_2 W_3\Lambda_3} \tag{4.170}$$

$$\begin{aligned} A_2^\infty = {} & \frac{1}{W_2\Lambda_2 W_3\Lambda_3}\left[W_2\Lambda_2(1 + 3W_3\Lambda_3 - W_1\Lambda_4 + W_1\Lambda_6)\right. \\ & \left. -2(W_2\Lambda_2)^2\left[1 - 2(W_1\Lambda_4 - W_1\Lambda_6)\right] - W_3\Lambda_3\right] \end{aligned} \tag{4.171}$$

$$A_3^\infty = \frac{(2W_2\Lambda_2 - 1)\left[W_3\Lambda_3 + 2W_2\Lambda_2(W_1\Lambda_4 - W_1\Lambda_6)\right]}{2W_2\Lambda_2 W_3\Lambda_3} \tag{4.172}$$

after imposing the second-order time accuracy conditions, Eq. (4.144), and the U0 conditions, Eq. (4.162). Therefore, equating Eq. (4.168) and Eq. (4.169), i.e.,

$$-A_1^\infty = \rho_\infty^{\min} + \rho_\infty^{\max} + \rho_\infty^{\mathrm{s}} \tag{4.173}$$

$$A_2^\infty = \rho_\infty^{\min}\rho_\infty^{\max} + \rho_\infty^{\max}\rho_\infty^{\mathrm{s}} + \rho_\infty^{\mathrm{s}}\rho_\infty^{\min} \tag{4.174}$$

$$-A_3^\infty = \rho_\infty^{\min}\rho_\infty^{\max}\rho_\infty^{\mathrm{s}} \tag{4.175}$$

leads to the following relations:

$$\begin{aligned}
W_1 &= 2W_2\Lambda_2 = \frac{1}{1+\rho_\infty^{\mathrm{s}}} \\
W_3\Lambda_3 &= \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \\
\Lambda_6 &= \frac{2+\rho_\infty^{\min}+\rho_\infty^{\max}+\rho_\infty^{\mathrm{s}}-\rho_\infty^{\min}\rho_\infty^{\max}\rho_\infty^{\mathrm{s}}}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \\
\lambda_1 &= \lambda_4 = 1, \;\; \Lambda_1 = \Lambda_4 = 1, \;\; \lambda_2 = \frac{1}{2} \\
\lambda_3 &= \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \\
\lambda_5 &= \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})}
\end{aligned} \tag{4.176}$$

Notice that at the low frequency limit, the above relation satisfies Eq. (4.155).

Algorithm 2

Single-field Form: I-GSSSS U0-Family of Algorithms for Linear Dynamical Systems

Integrator:

$$\begin{aligned}
&\left(\Lambda_6 W_1\mathbf{M} + \Lambda_5 W_2\mathbf{C}\Delta t + \Lambda_3 W_3\mathbf{K}\Delta t^2\right)\Delta\mathbf{a} \\
&= -\mathbf{M}\ddot{\mathbf{q}}_n - \mathbf{C}\left(\dot{\mathbf{q}}_n + \Lambda_4 W_1\ddot{\mathbf{q}}_n\Delta t\right) - \mathbf{K}\left(\mathbf{q}_n + \Lambda_1 W_1\dot{\mathbf{q}}_n\Delta t + \Lambda_2 W_2\ddot{\mathbf{q}}_n\Delta t^2\right) \\
&+ (1-W_1)\mathbf{f}_n + W_1\mathbf{f}_{n+1}
\end{aligned}$$

Updates:

$$\begin{aligned}
\mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + \lambda_3\Delta\mathbf{a}\Delta t^2 \\
\dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5\Delta\mathbf{a}\Delta t \\
\ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_n + \Delta\mathbf{a}
\end{aligned}$$

Initial conditions:

$$\begin{aligned}
\mathbf{q}(t_0) &= \mathbf{q}_0 \\
\dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0
\end{aligned}$$

Algorithmic parameters:

$$
\begin{aligned}
W_1\Lambda_1 &= \frac{1}{1+\rho_\infty^{\mathrm{s}}} \quad , \quad \lambda_1 = 1 \\
W_2\Lambda_2 &= \frac{1}{2(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_2 = \frac{1}{2} \\
W_3\Lambda_3 &= \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_3 = \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \\
W_1\Lambda_4 &= \frac{1}{1+\rho_\infty^{\mathrm{s}}} \quad , \quad \lambda_4 = 1 \\
W_2\Lambda_5 &= \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_5 = \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \\
W_1\Lambda_6 &= \frac{2+\rho_\infty^{\min}+\rho_\infty^{\max}+\rho_\infty^{\mathrm{s}}-\rho_\infty^{\min}\rho_\infty^{\max}\rho_\infty^{\mathrm{s}}}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})}
\end{aligned}
$$

The weighted time field is suggested to be

$$
W = 1 - \frac{15(1-2\rho_\infty^{\mathrm{s}})}{1-4\rho_\infty^{\mathrm{s}}}\frac{\tau}{\Delta t} + \frac{15(3-4\rho_\infty^{\mathrm{s}})}{1-4\rho_\infty^{\mathrm{s}}}\left(\frac{\tau}{\Delta t}\right)^2 - \frac{35(1-\rho_\infty^{\mathrm{s}})}{1-4\rho_\infty^{\mathrm{s}}}\left(\frac{\tau}{\Delta t}\right)^3
$$

In a general sense, they distinguish one algorithm design from another algorithm design; that is, no two algorithms can have the same DNA parameters, else, they are the same algorithm.

V0 Family of I-GSSSS Algorithms by Design

Following a similar procedure, we can also design the family of algorithms with the zero-order velocity overshoot within the I-GSSSS family of algorithms (***V0 Family*** of I-GSSSS algorithms) in terms of $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}})$. Imposing the second-order time accurate condition, i.e., Eq. (4.144), and the V0 condition, Eq. (4.164), leads to the following relations:

$$
\begin{aligned}
W_1 &= \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \\
\Lambda_6 &= \frac{2+\rho_\infty^{\min}+\rho_\infty^{\max}+\rho_\infty^{\mathrm{s}}-\rho_\infty^{\min}\rho_\infty^{\max}\rho_\infty^{\mathrm{s}}}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \\
W_2\Lambda_2 &= \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \\
W_3\Lambda_3 &= \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \\
\lambda_1 &= \lambda_4 = 1, \;\; \Lambda_1 = \Lambda_4 = 1, \;\; \lambda_2 = \frac{1}{2} \\
\lambda_3 &= \frac{1}{2(1+\rho_\infty^{\mathrm{s}})}, \;\; \lambda_5 = \frac{1}{1+\rho_\infty^{\mathrm{s}}}
\end{aligned}
\tag{4.177}
$$

Notice that at the low frequency limit, the above relation satisfies Eq. (4.155).

Algorithm 3

Single-field Form: I-GSSSS V0-Family of Algorithms for Linear Dynamical Systems

Integrator:

$$
\begin{aligned}
&\left(\Lambda_6 W_1\mathbf{M} + \Lambda_5 W_2\mathbf{C}\Delta t + \Lambda_3 W_3\mathbf{K}\Delta t^2\right)\Delta\mathbf{a} \\
&= -\mathbf{M}\ddot{\mathbf{q}}_n - \mathbf{C}\left(\dot{\mathbf{q}}_n + \Lambda_4 W_1\ddot{\mathbf{q}}_n\Delta t\right) - \mathbf{K}\left(\mathbf{q}_n + \Lambda_1 W_1\dot{\mathbf{q}}_n\Delta t + \Lambda_2 W_2\ddot{\mathbf{q}}_n\Delta t^2\right) \\
&+ (1-W_1)\mathbf{f}_n + W_1\mathbf{f}_{n+1}
\end{aligned}
$$

Updates:

$$
\begin{aligned}
\mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + \lambda_3\Delta\mathbf{a}\Delta t^2 \\
\dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5\Delta\mathbf{a}\Delta t \\
\ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_n + \Delta\mathbf{a}
\end{aligned}
$$

Initial conditions:

$$
\begin{aligned}
\mathbf{q}(t_0) &= \mathbf{q}_0 \\
\dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0
\end{aligned}
$$

Algorithmic parameters:

$$
\begin{aligned}
W_1\Lambda_1 &= \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \quad , \quad \lambda_1 = 1 \\
W_2\Lambda_2 &= \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \quad , \quad \lambda_2 = \frac{1}{2} \\
W_3\Lambda_3 &= \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_3 = \frac{1}{2(1+\rho_\infty^{\mathrm{s}})} \\
W_1\Lambda_4 &= \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \quad , \quad \lambda_4 = 1 \\
W_2\Lambda_5 &= \frac{2}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_5 = \frac{1}{1+\rho_\infty^{\mathrm{s}}} \\
W_1\Lambda_6 &= \frac{2+\rho_\infty^{\min}+\rho_\infty^{\max}+\rho_\infty^{\mathrm{s}}-\rho_\infty^{\min}\rho_\infty^{\max}\rho_\infty^{\mathrm{s}}}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})}
\end{aligned}
$$

The weighted time field is suggested to be

$$
\begin{aligned}
W =& 1 - \frac{30(3-4\rho_\infty^{\min}-4\rho_\infty^{\max}+6\rho_\infty^{\min}\rho_\infty^{\max})}{9-11\rho_\infty^{\min}-11\rho_\infty^{\max}+19\rho_\infty^{\min}\rho_\infty^{\max}}\frac{\tau}{\Delta t} \\
&+ \frac{15(25-37\rho_\infty^{\min}-37\rho_\infty^{\max}+53\rho_\infty^{\min}\rho_\infty^{\max})}{2(9-11\rho_\infty^{\min}-11\rho_\infty^{\max}+19\rho_\infty^{\min}\rho_\infty^{\max})}\left(\frac{\tau}{\Delta t}\right)^2 \\
&- \frac{35(3-5\rho_\infty^{\min}-5\rho_\infty^{\max}+7\rho_\infty^{\min}\rho_\infty^{\max})}{9-11\rho_\infty^{\min}-11\rho_\infty^{\max}+19\rho_\infty^{\min}\rho_\infty^{\max}}\left(\frac{\tau}{\Delta t}\right)^3
\end{aligned}
$$

Remark 7 (Algorithms 2 and 3)

1. *Algorithms 2 and 3 are the single step single solve family of algorithms which can be equivalently written in the form of the linear three-step method.*

2. *The algorithmic parameters in Algorithms 2 and 3 satisfy unique relations and serve as discrete numerically assigned [DNA] markers for any particular algorithm.*

3. *Various single step single solve algorithms commonly used in the research community at large belong to the single-field form I-GSSSS algorithmic framework; see Table 4.1 and Table 4.2. When U0/V0*$(1.0, 1.0, 1.0)$*, i.e.,* $\rho_\infty^{\min} = \rho_\infty^{\max} = \rho_\infty^{\mathrm{s}} = 1.0$ *in Algorithms 2 and 3, the recovered algorithm*

is called the classical ***midpoint rule with the endpoint acceleration (MRP-EPA)****; and when V0*$(1.0, 1.0, 0.0)$*, i.e.,* $\rho_\infty^{\min} = \rho_\infty^{\max} = 1.0$ *and* $\rho_\infty^{\mathrm{s}} = 0.0$ *in Algorithm 3, the recovered algorithm is a new* ***midpoint rule with the midpoint acceleration (MRP-MPA)****. The very important differences between the MPR-EPA and the MPR-MPA will be discussed later.*

4. *The weighting time fields* W *are determined from the spurious root at the high-frequency limit* ρ_∞^{s} *in Algorithm 2 and from the first principal root* $\rho_\infty^{\min}$ *and the second principal root* $\rho_\infty^{\max}$ *in Algorithm 3. The weighting time fields of the MPR-EPA (*U0/V0$(1.0, 1.0, 1.0)$*)/MPR-MPA (*V0$(1.0, 1.0, 0.0)$*), the implicit Newmark algorithm* U0$(1.0, 1.0, 0.0)$*, and an optimal scheme that is highly recommended, for example, with an arbitrary selection of parameter values* U0/V0$(0.8, 1.0, 0.8)$ *are given as follows:*

$$W(\tau) \;=\; 1 - 5\left(\frac{\tau}{\Delta t}\right) + 5\left(\frac{\tau}{\Delta t}\right)^2 \tag{4.178}$$

for the MPR-EPA/MPR-MPA;

$$W(\tau) \;=\; 1 - 15\left(\frac{\tau}{\Delta t}\right) + 45\left(\frac{\tau}{\Delta t}\right)^2 - 35\left(\frac{\tau}{\Delta t}\right)^3 \tag{4.179}$$

for the implicit Newmark algorithm; and

$$W(\tau) \;=\; 1 - \frac{45}{11}\left(\frac{\tau}{\Delta t}\right) + \frac{15}{11}\left(\frac{\tau}{\Delta t}\right)^2 + \frac{35}{11}\left(\frac{\tau}{\Delta t}\right)^3 \tag{4.180}$$

for the optimal scheme with typical spectral roots such as U0/V0$(0.8, 1.0, 0.8)$*; see Fig. 4.3. Note that the weighting time field for MPR-EPA or MPR-MPA is the same as the one for the midpoint rule in the two-field form as given in Eq.* (4.89)*; that is, the coefficient* w_3 *in Eq.* (4.110) *for MPR schemes, U0/V0(1.0,1.0,1.0) or V0(1.0,1.0,*$\rho_\infty^{\min}$*), vanishes, and the weighting function* W *reduces to the quadratic form in* τ*.*

5. *The total linear momentum in* $[t_n, t_{n+1}]$ *is exactly conserved, i.e.,* $\mathbf{L}_n = \mathbf{L}_{n+1}$*, for any spectral conditions* $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}})$*. The total energy (mechanical*

energy) in $[t_n, t_{n+1}]$ *is exactly conserved, i.e.,* $\mathscr{E}_n = \mathscr{E}_{n+1}$*, if* $\rho_\infty^{\min} = \rho_\infty^{\max} = 1.0$ *and* $\rho_\infty^{\mathrm{s}} \in [0.0, 1.0]$ *in both Algorithms 2 and 3. Notice that the implicit Newmark algorithm, i.e.,* $\rho_\infty^{\min} = \rho_\infty^{\max} = 1.0$ *and* $\rho_\infty^{\mathrm{s}} = 0.0$ *in Algorithm 2, is an energy-conserving algorithm only in linear dynamical systems.*

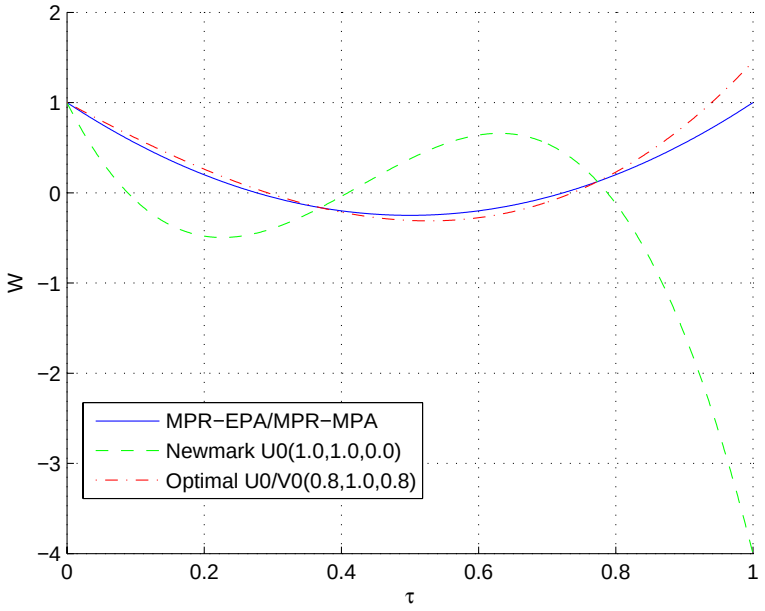


Figure 4.3: Comparisons of the weighting time fields for MPR-EPA/MPR-MPA, Newmark algorithm, and the optimal scheme (U0/V0$(0.8, 1.0, 0.8)$) ($\Delta t = 1.0$ sec)

Conservation of Mechanical Energy: Linear Dynamical Systems

We next show the spectral conditions which satisfy the conservation of mechanical energy within $[t_n, t_{n+1}]$ in the sense of $\mathscr{E}_n = \mathscr{E}_{n+1}$ in linear dynamical systems for Algorithms 2 and 3. Consider the mechanical energy given by $\mathscr{E}(\mathbf{q}, \dot{\mathbf{q}}) = \mathscr{T}(\dot{\mathbf{q}}) + \mathscr{U}(\mathbf{q})$ where $\mathscr{T}(\dot{\mathbf{q}}) = \frac{1}{2}\dot{\mathbf{q}} \cdot \mathbf{M}\dot{\mathbf{q}}$ and $\mathscr{U}(\mathbf{q}) = \frac{1}{2}\mathbf{q} \cdot \mathbf{K}\mathbf{q}$. We can write the difference of the kinetic energy within the time step as

$$\Delta\mathscr{T} := \mathscr{T}_{n+1} - \mathscr{T}_n = \Delta\mathbf{v} \cdot \mathbf{M}\dot{\mathbf{q}}_{n+1/2} \tag{4.181}$$

From the update, Eq. (4.117), we have

$$\Delta\mathbf{v} = \Delta t\left[\ddot{\mathbf{q}}_{n+W_1\Lambda_6} + (\lambda_5 - W_1\Lambda_6)\Delta\mathbf{a}\right] \tag{4.182}$$

Also, we can obtain the following relation, using Eq. (4.116):

$$\begin{aligned}\dot{\mathbf{q}}_{n+1/2} &= \dot{\mathbf{q}}_{n+\lambda_3/\lambda_5} + \left(\frac{1}{2} - \frac{\lambda_3}{\lambda_5}\right)\Delta\mathbf{v} \\ &= \frac{\Delta\mathbf{q}}{\Delta t} + \left(\frac{1}{2} - \frac{\lambda_3}{\lambda_5}\right)(\Delta\mathbf{v} - \Delta t\ddot{\mathbf{q}}_n) \\ &= \frac{\Delta\mathbf{q}}{\Delta t} + \Delta t\left(\frac{1}{2} - \frac{\lambda_3}{\lambda_5}\right)\lambda_5\Delta\mathbf{a}\end{aligned} \tag{4.183}$$

Substituting Eq. (4.182) and Eq. (4.183) into Eq. (4.181) gives

$$\begin{aligned}&\Delta\mathscr{T} + \left[\Delta\mathbf{q} + \Delta t^2\left(\frac{1}{2} - \frac{\lambda_3}{\lambda_5}\right)\lambda_5\Delta\mathbf{a}\right]\mathbf{K}\tilde{\mathbf{q}} \\ =&(\lambda_5 - W_1\Lambda_6)\left[\Delta\mathbf{q} + \Delta t^2\left(\frac{1}{2} - \frac{\lambda_3}{\lambda_5}\right)\lambda_5\Delta\mathbf{a}\right]\cdot\mathbf{M}\Delta\mathbf{a}\end{aligned} \tag{4.184}$$

using the relation, $\mathbf{M}\ddot{\mathbf{q}}_{n+W_1\Lambda_6} = -\mathbf{K}\tilde{\mathbf{q}}$ where

$$\begin{aligned}\tilde{\mathbf{q}} &= \mathbf{q}_{n+W_3} + \Delta t(W_1 - W_3)\dot{\mathbf{q}}_n \\ &= \mathbf{q}_{n+1/2} + \left(W_3 - \frac{1}{2}\right)\Delta\mathbf{q} + \Delta t(W_1 - W_3)\dot{\mathbf{q}}_n\end{aligned} \tag{4.185}$$

after some straightforward arrangements of Eq. (4.184), we obtain

$$\begin{aligned}\Delta\mathscr{E} &= \mathscr{E}_{n+1} - \mathscr{E}_n \\ &= \Delta\mathbf{q}\cdot\left[\mathbf{K}\left\{\left(\frac{1}{2} - W_3\right)\Delta\mathbf{q} + \Delta t(W_3 - W_1)\dot{\mathbf{q}}_n\right\} + (\lambda_5 - W_1\Lambda_6)\mathbf{M}\Delta\mathbf{a}\right] \\ &\quad + \Delta t\left(\frac{1}{2} - \frac{\lambda_3}{\lambda_5}\right)\lambda_5\Delta\mathbf{a}\cdot\mathbf{M}\ddot{\mathbf{q}}_{n+\lambda_5}\end{aligned} \tag{4.186}$$

If the right-hand side of Eq. (4.186) vanishes, the mechanical energy is exactly conserved within the time step.

Algorithm 2: In this framework, Eq. (4.186) can be reduced to

$$\begin{aligned}\Delta\mathscr{E} &= \Delta\mathbf{q}\cdot\left[\mathbf{K}\left(\frac{1}{2} - W_1\right)\Delta\mathbf{q} + (\lambda_5 - W_1\Lambda_6)\mathbf{M}\Delta\mathbf{a}\right] \\ &\quad + \Delta t\left(\frac{1}{2} - \frac{\lambda_3}{\lambda_5}\right)\lambda_5\Delta\mathbf{a}\cdot\mathbf{M}\ddot{\mathbf{q}}_{n+\lambda_5}\end{aligned} \tag{4.187}$$

The most relaxed spectral condition which satisfies $\Delta\mathscr{E} = 0$ is $\rho_\infty^{\max} = \rho_\infty^{\min} = 1$ with $\rho_\infty^{\mathrm{s}} \in [0, 1]$. That is, Eq. (4.187) can be written as

$$\Delta\mathscr{E} = \Delta\mathbf{q} \cdot [\mathbf{M}\Delta\mathbf{a} + \mathbf{K}\Delta\mathbf{q}] \left(\frac{1}{2} - W_1\right) = \mathbf{0} \tag{4.188}$$

where $\mathbf{M}\Delta\mathbf{a} + \mathbf{K}\Delta\mathbf{q} = \mathbf{0}$ since $\Lambda_6 = 1$ for $\rho_\infty^{\max} = \rho_\infty^{\min} = 1$. Notice that we do not need to additionally satisfy $W_1 = \frac{1}{2}$, i.e., $\rho_\infty^{\mathrm{s}} = 1$, to guarantee the conservation of mechanical energy in the linear dynamical system.

Algorithm 3: In this framework, Eq. (4.186) can be reduced to

$$\Delta\mathscr{E} = \Delta\mathbf{q} \cdot \left[\mathbf{K}\left\{\left(\frac{1}{2} - W_3\right)\Delta\mathbf{q} + \Delta t(W_3 - W_1)\dot{\mathbf{q}}_n\right\} + (\lambda_5 - W_1\Lambda_6)\mathbf{M}\Delta\mathbf{a}\right] \tag{4.189}$$

The most relaxed spectral condition for $\Delta\mathscr{E} = 0$ is $\rho_\infty^{\max} = \rho_\infty^{\min} = 1$ with $\rho_\infty^{\mathrm{s}} \in [0, 1]$, which gives $W_1 = W_3 = \frac{1}{2}$ and $\lambda_5 = W_1\Lambda_6 = \frac{1}{1+\rho_\infty^{\mathrm{s}}}$.

Measures of Accuracy: Numerical Dissipation and Numerical Dispersion

Consider the single-degree-of-freedom model problem with the initial conditions for a homogeneous system represented by

$$\begin{aligned} \ddot{q}(t) + 2\xi\omega\dot{q}(t) + \omega^2 q(t) &= 0 \\ q(0) = q_0 \ \text{ and } \ \dot{q}(0) &= \dot{q}_0 \end{aligned} \tag{4.190}$$

The exact solution of Eq. (4.190) at time t_n for the ***underdamped*** case ($0 \leq \xi < 1$) yields

$$q(t_n) = \exp\left[-\xi\Omega n\right]\left[\hat{c}_1^q \cos\left(\Omega_D n\right) + \hat{c}_2^q \sin\left(\Omega_D n\right)\right] \tag{4.191}$$

where $\omega_d := \omega\sqrt{1-\xi^2}$ (damped natural frequency), $\Omega_D := \omega_d \Delta t$, and the coefficients are given by

$$\begin{aligned} \hat{c}_1^q &= q_0 \\ \hat{c}_2^q &= \frac{\xi\omega q_0 + \dot{q}_0}{\omega_d} \end{aligned} \tag{4.192}$$

Similarly, the exact solutions for the ***critically-damped*** case ($\xi = 1$) and the ***overdamped*** case ($\xi > 1$) are given by

$$q(t_n) = \exp\left[-\Omega n\right]\left[q_0 + \left(\omega q_0 + \dot{q}_0\right) t_n\right] \tag{4.193}$$

$$q(t_n) = \exp\left[-\xi\Omega n\right]\left[q_0 \cosh\left(\hat{\Omega} n\right) + \frac{\xi\omega q_0 + \dot{q}_0}{\hat{\omega}} \sinh\left(\hat{\Omega} n\right)\right] \tag{4.194}$$

where $\hat{\omega} := \omega\sqrt{\xi^2 - 1}$ and $\hat{\Omega} := \hat{\omega}\Delta t$, respectively. Note that the eigenvalues of $\mathbf{A}$ are given as: two complex conjugate roots and one real root for the underdamped case; real and identical real roots for the critically-damped case; and real and distinct roots for the overdamped case. The implicit GSSSS family of algorithms are three-stage linear multistep methods, and can be written in the form

$$\mathbf{d}_{n+1} = \mathbf{A}\mathbf{d}_n \tag{4.195}$$

where $\mathbf{A} \in \mathbb{R}^{3\times 3}$ is the amplification matrix, and $\mathbf{d}_{n+1} = (q_{n+1}, \Delta t\dot{q}_{n+1}, \Delta t^2\ddot{q}_{n+1})^T$ and $\mathbf{d}_n = (q_n, \Delta t\dot{q}_n, \Delta t^2\ddot{q}_n)^T$. Hence, we have

$$\mathbf{d}_n = \mathbf{A}^n\mathbf{d}_0 \tag{4.196}$$

where n is the time step number. Suppose $\mathbf{A}$ has linearly independent eigenvectors. Then, the amplification matrix $\mathbf{A}$ satisfies

$$\mathbf{P}\mathbf{\Lambda} = \mathbf{A}\mathbf{P} \tag{4.197}$$

where $\mathbf{\Lambda} = \mathrm{diag}[\zeta_1, \zeta_2, \zeta_3]$ and $\mathbf{P}$ are the eigenvalue matrix and the eigenvector matrix of $\mathbf{A}$, and therefore, we have

$$\mathbf{A}^n = \mathbf{P}\mathbf{\Lambda}^n\mathbf{P}^{-1} \tag{4.198}$$

Hence, Eq. (4.196) yields

$$\mathbf{d}_n = \mathbf{P}\mathbf{\Lambda}^n\mathbf{P}^{-1}\mathbf{d}_0 \tag{4.199}$$

Assuming the eigenvalues $\{\zeta_i\}_{i=1}^3$ are distinct, i.e., $\zeta_1 \neq \zeta_2 \neq \zeta_3$, the discrete displacement at time t_n is given as

$$q_n = c_1^q\zeta_1^n + c_2^q\zeta_2^n + c_3^q\zeta_3^n \tag{4.200}$$

where the coefficients c_i^q $(i = 1, 2, 3)$ are determined from the initial conditions and the components of the eigenvectors. Similarly, the discrete velocity and acceleration at time t_n yields

$$\dot{q}_n = c_1^v \zeta_1^n + c_2^v \zeta_2^n + c_3^v \zeta_3^n \tag{4.201}$$

$$\ddot{q}_n = c_1^a \zeta_1^n + c_2^a \zeta_2^n + c_3^a \zeta_3^n \tag{4.202}$$

where the coefficients c_i^v and c_i^a $(i = 1, 2, 3)$ are also determined from the initial conditions and the components of the eigenvectors. If $\zeta_1 = \zeta_2$, the discrete displacement at time t_n are given as

$$q_n = (c_1^q + c_2^q n)\, \zeta_1^n + c_3^q \zeta_3^n \tag{4.203}$$

For $\zeta_1 = \zeta_2 = \zeta_3$,

$$q_n = \left(c_1^q + c_2^q n + c_3^q n^2\right) \zeta_1^n \tag{4.204}$$

Defining $\hat{\zeta} := \zeta - A_1/3$, Eq. (4.126) can be written as

$$\hat{\zeta}^3 + \frac{3A_2 - A_1^2}{3}\hat{\zeta} - \frac{2A_1^3 - 9A_1A_2 + 27A_3}{27} = 0 \tag{4.205}$$

If

$$\Delta := 3A_2 - A_1^2 + \left(\frac{2A_1^3 - 9A_1A_2 + 27A_3}{18}\right)^2 > 0 \tag{4.206}$$

the two complex conjugate eigenvalues are given as

$$\zeta_{1,2} = -\frac{\varsigma_1 + \varsigma_2}{2} \pm i\frac{3(\varsigma_1 - \varsigma_2)}{2} \tag{4.207}$$

and the real eigenvalue is given as

$$\zeta_3 = \varsigma_1 + \varsigma_2 \tag{4.208}$$

where

$$\varsigma_1 = \left(\frac{2A_1^3 - 9A_1A_2 + 27A_3}{54} + \sqrt{\Delta}\right)^{1/3} \tag{4.209}$$

$$\varsigma_2 = \left(\frac{2A_1^3 - 9A_1A_2 + 27A_3}{54} - \sqrt{\Delta}\right)^{1/3} \tag{4.210}$$

Suppose that the characteristic equation yields two complex conjugate roots ($\zeta_{1,2} = a \pm ib$) and one real root ($\zeta_3 \in \mathbb{R}$), assuming $0 \leq |\zeta_3| = \rho_\infty^{\mathrm{s}} < |\zeta_{1,2}| = \rho_\infty^{\min} = \rho_\infty^{\max} \leq 1$. In this case, the spectral radius is defined as the modulus of $\zeta_{1,2}$,

$$\rho := |\zeta_{1,2}| = \sqrt{a^2 + b^2} > |\zeta_3| \tag{4.211}$$

where

$$a = \frac{A_1}{3} - \frac{\varsigma_1 + \varsigma_2}{2} \tag{4.212}$$

$$b = \frac{(\varsigma_1 - \varsigma_2)\sqrt{3}}{2} \tag{4.213}$$

and the discrete solution of Eq. (4.200) can be written in the form

$$q_n = \exp\left[-\bar{\xi}\bar{\Omega}n\right]\left[\bar{c}_1^q \cos\left(\bar{\Omega}_D n\right) + \bar{c}_2^q \sin\left(\bar{\Omega}_D n\right)\right] + c_3^q \zeta_c^n \tag{4.214}$$

where

$$\begin{aligned} \bar{\Omega} &= \frac{\bar{\Omega}_D}{\sqrt{1-\bar{\xi}^2}} = \bar{\omega}\Delta t, \quad \bar{\Omega}_D = \tan^{-1}\left(\frac{b}{a}\right) \\ \bar{\xi} &= -\frac{\ln(a^2+b^2)}{2\bar{\Omega}} = -\frac{\ln(\rho)}{\bar{\Omega}} \end{aligned} \tag{4.215}$$

Note that $\bar{\xi}$ is called the ***algorithmic damping*** or ***numerical dissipation***. The ***relative period error*** or ***numerical dispersion*** is defined as

$$P_{nd} = \frac{\bar{T} - T}{T} = \frac{\Omega}{\bar{\Omega}} - 1 \tag{4.216}$$

where

$$\bar{T} = \frac{2\pi}{\bar{\omega}} \quad T = \frac{2\pi}{\omega} \tag{4.217}$$

Time Level Consistency of the Equation of Motion

Both Algorithm 2 and Algorithm 3 have been particularly designed to be of second-order time accuracy in $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\mathbf{f}$; however, we get only first-order accuracy if we assume $\ddot{\mathbf{q}}_n \approx \ddot{\mathbf{q}}(t_n)$ and $\ddot{\mathbf{q}}_{n+1} \approx \ddot{\mathbf{q}}(t_{n+1})$. In order to guarantee the second-order time accuracy not only in $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\mathbf{f}$, but also in $\ddot{\mathbf{q}}$ in Algorithm 2 and Algorithm 3, one must be aware of the correct time level of approximations for $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\ddot{\mathbf{q}}$ as

$$\begin{aligned} \mathbf{q}_n &\approx \mathbf{q}(t_n) \;\text{ and }\; \mathbf{q}_{n+1} \approx \mathbf{q}(t_{n+1}) \\ \dot{\mathbf{q}}_n &\approx \dot{\mathbf{q}}(t_n) \;\text{ and }\; \dot{\mathbf{q}}_{n+1} \approx \dot{\mathbf{q}}(t_{n+1}) \\ \ddot{\mathbf{q}}_n &\approx \ddot{\mathbf{q}}(t_{n-\phi}) \;\text{ and }\; \ddot{\mathbf{q}}_{n+1} \approx \ddot{\mathbf{q}}(t_{n+1-\phi}) \end{aligned} \tag{4.218}$$

where $\phi := W_1(\Lambda_6 - 1) \in \mathbb{R}$.

Theorem 4 (Time Level of Acceleration)
In Algorithm 2 and Algorithm 3, the second-order time accuracy in $\ddot{\mathbf{q}}$*, i.e.,*

$$\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}(t_{n+1-\phi}) = \mathcal{O}(\Delta t^2) \;\; \textit{with} \;\; \phi := W_1(\Lambda_6 - 1) \tag{4.219}$$

is obtained for a solution satisfying

$$\begin{aligned} \mathbf{q}_n - \mathbf{q}(t_n) = \mathcal{O}(\Delta t^2) \;\; \textit{and} \;\; \mathbf{q}_{n+1} - \mathbf{q}(t_{n+1}) = \mathcal{O}(\Delta t^2) \\ \dot{\mathbf{q}}_n - \dot{\mathbf{q}}(t_n) = \mathcal{O}(\Delta t^2) \;\; \textit{and} \;\; \dot{\mathbf{q}}_{n+1} - \dot{\mathbf{q}}(t_{n+1}) = \mathcal{O}(\Delta t^2) \end{aligned} \tag{4.220}$$

if $\ddot{\mathbf{q}}_n - \ddot{\mathbf{q}}(t_{n-\phi}) = \mathcal{O}(\Delta t^2)$ *is guaranteed.*

Proof. In this proof, we only consider Algorithm 3 since the proof for Algorithm 2 is more straightforward, and we get the same result as stated in the theorem above. Algorithm 3 can cast into

$$\mathbf{M}\tilde{\mathbf{a}} + \mathbf{C}\tilde{\mathbf{v}} + \mathbf{K}\tilde{\mathbf{q}} = \tilde{\mathbf{f}} \tag{4.221}$$

where the algorithmic $\ddot{\mathbf{q}}$, $\dot{\mathbf{q}}$, $\mathbf{q}$, and $\mathbf{f}$ are given by

$$\begin{aligned}
\tilde{\mathbf{a}} &= (1 - W_1\Lambda_6)\ddot{\mathbf{q}}_n + W_1\Lambda_6\ddot{\mathbf{q}}_{n+1} && (4.222)\\
\tilde{\mathbf{v}} &= \dot{\mathbf{q}}_n + W_2(\dot{\mathbf{q}}_{n+1} - \dot{\mathbf{q}}_n) + \Delta t(W_1 - W_2)\ddot{\mathbf{q}}_n && (4.223)\\
\tilde{\mathbf{q}} &= \mathbf{q}_n + W_2(\mathbf{q}_{n+1} - \mathbf{q}_n) + \Delta t(W_1 - W_2)\dot{\mathbf{q}}_n && (4.224)\\
\tilde{\mathbf{f}} &= (1 - W_1)\mathbf{f}_n + W_1\mathbf{f}_{n+1} && (4.225)
\end{aligned}$$

Using Equation (4.220), we get

$$\begin{aligned}
\tilde{\mathbf{v}} &= \dot{\mathbf{q}}(t_n) + W_2[\dot{\mathbf{q}}(t_{n+1}) - \dot{\mathbf{q}}(t_n)] + \Delta t(W_1 - W_2)\ddot{\mathbf{q}}(t_{n-\phi}) + \mathcal{O}(\Delta t^p)\\
&= \dot{\mathbf{q}}(t_n) + \Delta t W_1\ddot{\mathbf{q}}(t_n) + \mathcal{O}(\Delta t^p) && (4.226)
\end{aligned}$$

where $p = 2$ and $p = 1$ for $s \geq 2$ and $s = 1$, respectively, in $\ddot{\mathbf{q}}_n - \ddot{\mathbf{q}}(t_{n-\phi}) = \mathcal{O}(\Delta t^s)$. Similarly,

$$\begin{aligned}
\tilde{\mathbf{q}} &= \mathbf{q}(t_n) + W_2[\mathbf{q}(t_{n+1}) - \mathbf{q}(t_n)] + \Delta t(W_1 - W_2)\dot{\mathbf{q}}(t_{n-\phi}) + \mathcal{O}(\Delta t^2)\\
&= \mathbf{q}(t_n) + \Delta t W_1\dot{\mathbf{q}}(t_n) + \mathcal{O}(\Delta t^2) && (4.227)
\end{aligned}$$

and

$$\tilde{\mathbf{f}} = \mathbf{f}(t_n) + W_1[\mathbf{f}(t_{n+1}) - \mathbf{f}(t_n)] + \mathcal{O}(\Delta t^2) = \mathbf{f}(t_n) + \Delta t W_1\dot{\mathbf{f}}(t_n) + \mathcal{O}(\Delta t^2) \quad (4.228)$$

Since the Taylor series expansions about time $t = t_n$ yields

$$\begin{aligned}
\ddot{\mathbf{q}}(t_{n+1-\phi}) &= \ddot{\mathbf{q}}(t_n) + (1 - \phi)\Delta t\dddot{\mathbf{q}}(t_n) + \mathcal{O}(\Delta t^2)\\
\ddot{\mathbf{q}}(t_{n-\phi}) &= \ddot{\mathbf{q}}(t_n) - \phi\Delta t\dddot{\mathbf{q}}(t_n) + \mathcal{O}(\Delta t^2)
\end{aligned} \quad (4.229)$$

we get

$$(1 - W_1\Lambda_6)\ddot{\mathbf{q}}(t_{n-\phi}) + W_1\Lambda_6\ddot{\mathbf{q}}(t_{n+1-\phi}) = \ddot{\mathbf{q}}(t_n) + \Delta t(W_1\Lambda_6 - \phi)\dddot{\mathbf{q}}(t_n) + \mathcal{O}(\Delta t^2) \quad (4.230)$$

From Equations (4.221)-(4.230) with $\ddot{\mathbf{q}}(t) = -\mathbf{M}^{-1}[\mathbf{C}\dot{\mathbf{q}}(t) + \mathbf{K}\mathbf{q}(t) - \mathbf{f}(t)]$,

$$\begin{aligned}
&(1 - W_1\Lambda_6)[\ddot{\mathbf{q}}_n - \ddot{\mathbf{q}}(t_{n-\phi})] + W_1\Lambda_6[\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}(t_{n+1-\phi})]\\
=&\Delta t(W_1 - W_1\Lambda_6 + \phi)\dddot{\mathbf{q}}(t_n) + \mathcal{O}(\Delta t^p)
\end{aligned} \quad (4.231)$$

Hence, $\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}(t_{n+1-\phi}) = \mathcal{O}(\Delta t^2)$ is obtained when $\ddot{\mathbf{q}}_n - \ddot{\mathbf{q}}(t_{n-\phi}) = \mathcal{O}(\Delta t^s)$ with $s \geq 2$ for $\phi = W_1(\Lambda_6 - 1)$. ■

Remark 8 (Theorem 4)

1. *Choosing the initial value of* $\ddot{\mathbf{q}}$ *as* $\ddot{\mathbf{q}}_0 = \ddot{\mathbf{q}}(t_0) = -\mathbf{M}^{-1}[\mathbf{C}\dot{\mathbf{q}}(t_0) + \mathbf{K}\mathbf{q}(t_0) - \mathbf{f}(t_0)]$, *actually still gives the second-order accuracy in* $\ddot{\mathbf{q}}$.

2. *It is important to note that* $\ddot{\mathbf{q}}_n \approx \ddot{\mathbf{q}}(t_{n-\phi})$ *and* $\ddot{\mathbf{q}}_{n+1} \approx \ddot{\mathbf{q}}(t_{n+1-\phi})$ *are not the approximations of* $\ddot{\mathbf{q}}$ *at time* $t = t_n$ *and* $t = t_{n+1}$, *respectively. Therefore, to plot the* ***true acceleration*** *time history, one must use*

$$\ddot{\mathbf{q}}(t_1) \approx \frac{1}{1-\phi}\ddot{\mathbf{q}}_1 - \frac{\phi}{1-\phi}\ddot{\mathbf{q}}_0 \tag{4.232}$$

at the first time step and

$$\ddot{\mathbf{q}}(t_{n+1}) \approx (1+\phi)\ddot{\mathbf{q}}_{n+1} - \phi\ddot{\mathbf{q}}_n \tag{4.233}$$

for $n \in \{1, 2, \cdots, f-1\}$.

3. *The spectral condition V0*$\{1.0, 1.0, \rho_\infty^s\}$ *with* $\rho_\infty^s \in [0.0, 1.0]$ *in Algorithm 3 leads to*

$$\mathbf{M}\tilde{\mathbf{a}} + \mathbf{C}\left[\dot{\mathbf{q}}_n + \frac{\Delta t}{2}\tilde{\mathbf{a}}\right] + \mathbf{K}\left[\mathbf{q}_n + \frac{\Delta t}{2}\dot{\mathbf{q}}_n + \frac{\Delta t^2}{4}\tilde{\mathbf{a}}\right] = \tilde{\mathbf{f}} \tag{4.234}$$

with

$$\begin{aligned} \mathbf{q}_{n+1} &= \mathbf{q}_n + \Delta t\dot{\mathbf{q}}_n + \frac{\Delta t}{2}\tilde{\mathbf{a}} \\ \dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \Delta t\tilde{\mathbf{a}} \end{aligned} \tag{4.235}$$

where

$$\tilde{\mathbf{a}} = \ddot{\mathbf{q}}_n + \frac{1}{1+\rho_\infty^s}(\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n) \tag{4.236}$$

Note that we have

$$W_1 = \frac{1}{2} \;\; \textit{and} \;\; \phi = \frac{1-\rho_\infty^s}{2(1+\rho_\infty^s)} \in [0, \tfrac{1}{2}] \tag{4.237}$$

for V0:$\{(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}}) = (1.0, 1.0, \rho_\infty^{\mathrm{s}})\}$ *with* $\rho_\infty^{\mathrm{s}} \in [0.0, 1.0]$. *We call U0/V0:*$\{(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}}) = (1.0, 1.0, 1.0)\}$, *i.e.,* $\ddot{\mathbf{q}}_{n+1} \approx \ddot{\mathbf{q}}(t_{n+1})$, *and V0:*$\{(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}}) = (1.0, 1.0, 0.0)\}$, *i.e.,* $\ddot{\mathbf{q}}_{n+1} \approx \ddot{\mathbf{q}}(t_{n+1/2})$, *the* ***midpoint rule with the endpoint acceleration (MRP-EPA)****, which is the classical midpoint rule, and the new* ***midpoint rule with the midpoint acceleration (MRP-MPA)****, respectively. Notice that the difference between MPR-EPA and MPR-MPA is*

$$\frac{\dot{\mathbf{q}}_{n+1} - \dot{\mathbf{q}}_n}{\Delta t} = \tilde{\mathbf{a}} = \begin{cases} \ddot{\mathbf{q}}_{n+1/2} \approx \frac{1}{2}[\ddot{\mathbf{q}}(t_{n+1}) + \ddot{\mathbf{q}}(t_n)] & \text{for MPR-EPA} \\ \ddot{\mathbf{q}}_{n+1} \approx \ddot{\mathbf{q}}(t_{n+1/2}) & \text{for MPR-MPA} \end{cases} \tag{4.238}$$

Note that the information regarding the acceleration at the previous time step is not required to evaluate the current acceleration, and consequently, the new MPR-MPA provides a robust computational algorithm for general dynamics applications. By eliminating the acceleration, Eq. (4.234)-*Eq.*(4.236) *can be reduced to the classical symplectic midpoint rule which takes the form*

$$\begin{aligned} \mathbf{M}\frac{\dot{\mathbf{q}}_{n+1} - \dot{\mathbf{q}}_n}{\Delta t} + \mathbf{C}\frac{\dot{\mathbf{q}}_{n+1} + \dot{\mathbf{q}}_n}{2} + \mathbf{K}\frac{\mathbf{q}_{n+1} + \mathbf{q}_n}{2} &= \tilde{\mathbf{f}} \\ \frac{\dot{\mathbf{q}}_{n+1} + \dot{\mathbf{q}}_n}{2} &= \frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t} \end{aligned} \tag{4.239}$$

where $\tilde{\mathbf{f}} = \mathbf{f}(t_{n+1/2})$ *can be used for the algorithmic time-dependent external force instead of* $\tilde{\mathbf{f}} = (\mathbf{f}_n + \mathbf{f})/2$ *since* $\mathbf{f}(t_{n+\alpha}) - \mathbf{f}_{n+\alpha} = \mathcal{O}(\Delta t^2)$ *for* $\alpha \in [0, 1]$.

Theorem 5 (Time Level Consistency)
In order to guarantee the second-order time accuracy in all time-dependent variables, such as $\mathbf{q}(t)$, $\dot{\mathbf{q}}(t)$, $\ddot{\mathbf{q}}(t)$, *and* $\mathbf{f}(t)$, *in the discrete equation of motion, they must be evaluated at the same time level* $t = t^* = t_{n+W_1}$ *as*

$$\mathbf{0} = \mathbf{M}\tilde{\mathbf{a}} + \mathbf{C}\tilde{\mathbf{v}} + \mathbf{K}\tilde{\mathbf{q}} - \tilde{\mathbf{f}} + \mathcal{O}(\Delta t^2) = \mathbf{M}\ddot{\mathbf{q}}(t^*) + \mathbf{C}\dot{\mathbf{q}}(t^*) + \mathbf{K}\mathbf{q}(t^*) - \mathbf{f}(t^*) \tag{4.240}$$

Proof. By using the Taylor series expansions about time $t = t_n$, we get

$$\begin{aligned} \tilde{\mathbf{a}} &= (1 - W_1\Lambda_6)\ddot{\mathbf{q}}(t_{n-\phi}) + W_1\Lambda_6\ddot{\mathbf{q}}(t_{n+1-\phi}) + \mathcal{O}(\Delta t^2) \\ &= \ddot{\mathbf{q}}(t_n) + \Delta t W_1 \dddot{\mathbf{q}}(t_n) + \mathcal{O}(\Delta t^2) \\ &= \ddot{\mathbf{q}}(t_{n+W_1}) \end{aligned} \tag{4.241}$$

when $\ddot{\mathbf{q}}_n - \ddot{\mathbf{q}}(t_{n-\phi}) = \mathcal{O}(\Delta t^2)$. Similarly, we get

$$\tilde{\mathbf{v}} = \dot{\mathbf{q}}(t_n) + \Delta t W_1 \ddot{\mathbf{q}}(t_n) + \mathcal{O}(\Delta t^2) = \dot{\mathbf{q}}(t_{n+W_1}) \tag{4.242}$$

$$\tilde{\mathbf{q}} = \mathbf{q}(t_n) + \Delta t W_1 \dot{\mathbf{q}}(t_n) + \mathcal{O}(\Delta t^2) = \mathbf{q}(t_{n+W_1}) \tag{4.243}$$

$$\tilde{\mathbf{f}} = \mathbf{f}(t_n) + \Delta t W_1 \dot{\mathbf{f}}(t_n) + \mathcal{O}(\Delta t^2) = \mathbf{f}(t_{n+W_1}) \tag{4.244}$$

Therefore, the discrete equation of motion is evaluated at the time level $t = t_{n+W_1}$ when the second-order time accuracy in $\mathbf{q}$, $\dot{\mathbf{q}}$, $\ddot{\mathbf{q}}$, and $\mathbf{f}$ are guaranteed. The time level $t^* = t_{n+W_1}$ is called the ***algorithmic time level***. If the algorithmic balance equation is satisfied at the time level $t^* = t_{n+W_1}$ with the error $\mathcal{O}(\Delta t^2)$, the acceleration, velocity, and configuration are of second-order time accuracy. The algorithmic time level consistency in the balance equation is illustrated in Fig. 4.4. ■

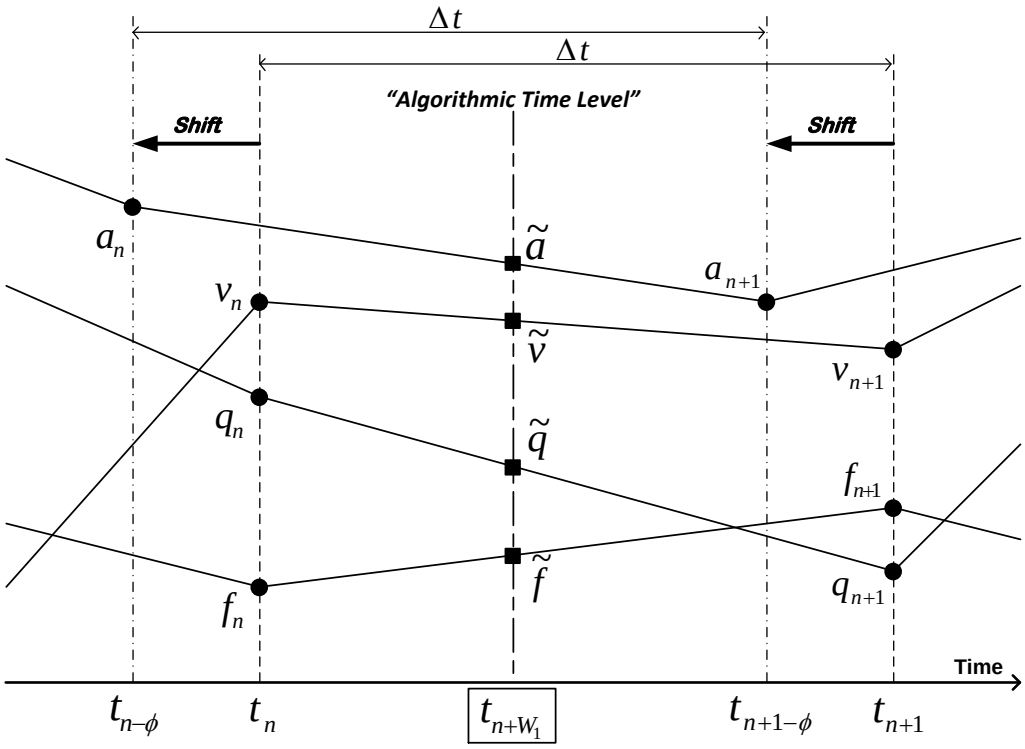


Figure 4.4: Illustration of the algorithmic time level

Algorithm 2	**Spectral Conditions**	**Algorithm 3**
MPR-EPA	$\rho_\infty^{\min} = \rho_\infty^{\max} = \rho_\infty^{\mathrm{s}} = 1$	***MPR-EPA***
Implicit Newmark [33]	$\rho_\infty^{\min} = \rho_\infty^{\max} = 1,\ \rho_\infty^{\mathrm{s}} = 0$	***MPR-MPA***
$\mathrm{U0V0}_{\mathrm{Optimal}}$ [34]	$\rho_\infty^{\min} = \rho_\infty^{\mathrm{s}} \in [0,1],\ \rho_\infty^{\max} = 1$	$\mathrm{V0U0}_{\mathrm{Optimal}}$ [34]
$\mathrm{U0V0}_{\mathrm{DA}}$ [34]	$\rho_\infty^{\min} \in [0,1],\ \rho_\infty^{\max} = 1,\ \rho_\infty^{\mathrm{s}} = 0$	$\mathrm{V0U0}_{\mathrm{DA}}$ [34]
$\mathrm{U0V0}_{CA}$ [34]	$\rho_\infty^{\min} \in [0,1],\ \rho_\infty^{\max} = 1$ $\rho_\infty^{\mathrm{s}} = \frac{1-\rho_\infty^{\min}\rho_\infty^{\max}}{\rho_\infty^{\min}+\rho_\infty^{\max}+2\rho_\infty^{\min}\rho_\infty^{\max}}$	$\mathrm{V0U0}_{CA}$ [34]
U0V0	$\rho_\infty^{\min} \in [0,1] \neq \rho_\infty^{\mathrm{s}},\ \rho_\infty^{\max} = 1$ $0 \neq \rho_\infty^{\mathrm{s}} \neq \frac{1-\rho_\infty^{\min}\rho_\infty^{\max}}{\rho_\infty^{\min}+\rho_\infty^{\max}+2\rho_\infty^{\min}\rho_\infty^{\max}}$	V0U0
$\mathrm{U0V1}_{\mathrm{Optimal}}$ [34] Three Parameter Optimal [4, 5]	$\rho_\infty^{\min} = \rho_\infty^{\max} = \rho_\infty^{\mathrm{s}} \in [0,1]$	$\mathrm{V0U1}_{\mathrm{Optimal}}$
$\mathrm{U0V1}_{\mathrm{DA}}$ [34] WBZ [35]	$\rho_\infty^{\min} = \rho_\infty^{\max} \in [0,1],\ \rho_\infty^{\mathrm{s}} = 0$	$\mathrm{V0U1}_{\mathrm{DA}}$
$\mathrm{U0V1}_{\mathrm{CA}}$ [34] HHT-α [21]	$\rho_\infty^{\min} = \rho_\infty^{\max} \in [0.5, 1.0]$ $\rho_\infty^{\mathrm{s}} = \frac{1-\rho_\infty^{\min}\rho_\infty^{\max}}{\rho_\infty^{\min}+\rho_\infty^{\max}+2\rho_\infty^{\min}\rho_\infty^{\max}}$	$\mathrm{V0U1}_{\mathrm{CA}}$
U0V1	$\rho_\infty^{\min} \neq \rho_\infty^{\mathrm{s}},\ \rho_\infty^{\min} = \rho_\infty^{\max} \in [0,1]$ $0 \neq \rho_\infty^{\mathrm{s}} \neq \frac{1-\rho_\infty^{\min}\rho_\infty^{\max}}{\rho_\infty^{\min}+\rho_\infty^{\max}+2\rho_\infty^{\min}\rho_\infty^{\max}}$	V0U1
$\mathrm{U0V1}_{\mathrm{Optimal}}$	$\rho_\infty^{\max} \in [0,1] \neq \rho_\infty^{\min}$ $\rho_\infty^{\min} = \rho_\infty^{\mathrm{s}} \in [0,1]$	$\mathrm{V0U1}_{\mathrm{Optimal}}$
$\mathrm{U0V1}_{\mathrm{DA}}$	$\rho_\infty^{\min} \in [0,1],\ \rho_\infty^{\max} \in [0,1]$ $\rho_\infty^{\min} \neq \rho_\infty^{\max},\ \rho_\infty^{\mathrm{s}} = 0$	$\mathrm{V0U1}_{\mathrm{DA}}$
$\mathrm{U0V1}_{\mathrm{CA}}$	$\rho_\infty^{\min} \in [0,1] \neq \rho_\infty^{\max} \in [0,1]$ $\rho_\infty^{\mathrm{s}} = \frac{1-\rho_\infty^{\min}\rho_\infty^{\max}}{\rho_\infty^{\min}+\rho_\infty^{\max}+2\rho_\infty^{\min}\rho_\infty^{\max}}$	$\mathrm{V0U1}_{\mathrm{CA}}$
U0V1	$\rho_\infty^{\min} \neq \rho_\infty^{\mathrm{s}},\ \rho_\infty^{\min} \neq \rho_\infty^{\max}$ $0 \neq \rho_\infty^{\mathrm{s}} \neq \frac{1-\rho_\infty^{\min}\rho_\infty^{\max}}{\rho_\infty^{\min}+\rho_\infty^{\max}+2\rho_\infty^{\min}\rho_\infty^{\max}}$	V0U1

Table 4.1: Various U0-family and V0-family of algorithms within the single-field form I-GSSSS algorithmic framework [2] encompassing LMS methods. U0-$\mathrm{V0}_{\mathrm{Optimal}}$ are the recommended optimal algorithms with controllable numerical dissipation. MPR-MPA is the recommended algorithm without numerical dissipation.

Algorithms	$W_1\Lambda_1$	$W_2\Lambda_2$	$W_3\Lambda_3$	$W_1\Lambda_4$	$W_2\Lambda_5$	$W_1\Lambda_6$
Newmark-β [33]	1	1/2	β	1	γ	1
Average Acceleration	1	1/2	1/4	1	1/2	1
Linear Acceleration	1	1/2	1/6	1	1/2	1
Fox-Goodwin	1	1/2	1/12	1	1/2	1
Central Difference	1	1/2	0	1	1/2	1
Wilson-θ [36]	θ	$\theta^2/2$	$\theta^3/6$	θ	$\theta^2/2$	θ
HHT-α [21]	$1+\alpha$	$(1+\alpha)/2$	$(1-\alpha)(1-\alpha^2)/4$	$1+\alpha$	$(1+\alpha)(1/2-\alpha)$	1
Hoff-Pahl	θ_1	1/2	$\theta_1^2/4$	θ_1	$3/2-\theta_1$	1
Bossak-Newmark	1	1/2	$(1-\alpha_\beta)^2/4$	1	$1/2-\alpha_\beta$	$1-\alpha_\beta$
TPO [4, 5]	$1-\alpha_f$	$(1-\alpha_f)/2$	$(1-\alpha_f)(1-\alpha_m-\alpha_f)/4$	$1-\alpha_f$	$(1-\alpha_f)(1/2-\alpha_m+\alpha_f)$	$1-\alpha_m$

Algorithms	λ_1	λ_2	λ_3	λ_4	λ_5
Newmark-β	1	1/2	β	1	γ
Average Acceleration	1	1/2	1/4	1	1/2
Linear Acceleration	1	1/2	1/6	1	1/2
Fox-Goodwin	1	1/2	1/12	1	1/2
Central Difference	1	1/2	0	1	1/2
Wilson-θ	1	1/2	1/6	1	1/2
HHT-α	1	1/2	$(1-\alpha)^2/4$	1	$1/2-\alpha$
Hoff-Pahl	1	1/2	$\theta_1^2/4$	1	$3/2-\theta_1$
Bossak-Newmark	1	1/2	$(1-\alpha_\beta)^2/4$	1	$1/2-\alpha_\beta$
TPO	1	1/2	$(1-\alpha_m-\alpha_f)^2/4$	1	$1/2-\alpha_m+\alpha_f$

Table 4.2: The relationship of the algorithmic parameters between the single-field form I-GSSSS U0-family of algorithms and several classical direct time integration schemes. "TPO" above stands for the three parameter optimal method, and note that the Generalized-α method [3] is identical (see discussion in V0 family earlier).

4.3 Numerical Illustrations of the Family of the Implicit GSSSS algorithms

The plots of the time accuracy, stability, mechanical energy, numerical dissipation, and numerical dispersion of selected schemes within Algorithm 2, U0 $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}})$, and Algorithm 3, V0$(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}})$, are shown in this section. Note that we also discuss algorithms with first-order displacement and velocity overshoot behaviors in addition to the zero-order displacement and velocity overshoot behaviors. Higher order overshoot algorithms are not shown here or discussed as they are not competitive.

Time Accuracy: The time accuracy plots of the selected algorithms within Algorithms 2 and Algorithm 3 are shown in Fig 4.6. Fig 4.6 shows that the order of time accuracy of the configuration ($\square$), velocity ($\triangle$), and acceleration ($\circ$) is 2 for all the schemes shown. To plot the time accuracy, we follow the steps shown below:

`Step 0:`

Input ξ, ω, and the time-dependent external load $g(t)$ for the modal SDOF equation $\ddot{q} + 2\xi\omega\dot{q} + \omega^2 q = g(t)$; initial conditions $\mathbf{q}_0 = \mathbf{q}(t_0)$ and $\dot{\mathbf{q}}_0 = \dot{\mathbf{q}}(t_0)$ at $n = 0$; the principal and spurious roots, $\rho_\infty^{\min}$, $\rho_\infty^{\max}$, and ρ_∞^{s}; and some time step sizes $\Delta t_a > 0$ for $a = 1, 2, 3, \cdots, k$.

`Step 1:`

Obtain the algorithmic parameters from $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}})$.

Obtain the initial acceleration from

$$\ddot{q}_0 = -2\xi\omega\dot{q}_0 - \omega^2 q_0 + g(t_0) \tag{4.245}$$

`Step 2:`

Correct the values of the configuration, velocity, and acceleration at $n = n^*$.

`Step 3:`

Plot the errors in the configuration, velocity, and acceleration at $n = n^*$, defined by

$$\text{Error}_q := \log_{10}\left|\frac{q_{n^*} - q(t_{n^*})}{q(t_{n^*})}\right| \tag{4.246}$$

$$\text{Error}_v := \log_{10}\left|\frac{\dot{q}_{n^*} - \dot{q}(t_{n^*})}{\dot{q}(t_{n^*})}\right| \tag{4.247}$$

$$\text{Error}_a := \log_{10}\left|\frac{a'_{n^*} - \ddot{q}(t_{n^*})}{\ddot{q}(t_{n^*})}\right| \tag{4.248}$$

respectively against $\log_{10}\Omega_a$ where $q(t_{n^*})$, $\dot{q}(t_{n^*})$, and $\ddot{q}(t_{n^*})$ are the exact configuration, velocity, and acceleration at time t_{n^*}. Note that a'_{n^*} satisfies $a'_{n^*} - \ddot{q}(t_{n^*}) = \mathcal{O}(\Delta t^2)$.

Selected Algorithms (in Fig. 4.6):

U0V0/V0U0$(1.0, 1.0, 1.0)$: MPR-EPA (classical midpoint rule)

U0V0$(1.0, 1.0, 0.0)$: Implicit Newmark

V0U0$(1.0, 1.0, 0.0)$: MPR-MPA (new midpoint rule)

U0V0/V0U0$(0.8, 1.0, 0.8)$: U0V0/V0U0 Optimal

U0V1$(0.6, 0.9, 0.2)$

V0U1$(0.6, 0.9, 0.2)$

Stability (Spectral Plot): The spectral plots of the selected algorithms within Algorithms 2 and Algorithm 3 are shown in Fig 4.7. In Fig 5.5, $\rho_\infty^{\min}$, $\rho_\infty^{\max}$ (spectral radius), and ρ_∞^{s} (spurious root) are denoted by blue-colored curves with $\square$, red-colored curves with $\triangle$, and black-colored curves with $\circ$, respectively. To plot the spectral curves, follow the steps shown below:

`Step 0:`

Input sample frequencies $\Omega_a := \omega\Delta t_a > 0$ for $a = 1, 2, 3, \cdots, k$.

`Step 1:`

Find the eigenvalues, $\{\zeta\}_{i=1}^{3}$, of the amplification matrices $\mathbf{A} \in \mathbb{R}^{3\times 3}$, and define the spectral radius and spurious root,

$$\rho_2 := \max\left(\left|\{\zeta_i\}_{i=1}^{3}\right|\right) \equiv \rho \tag{4.249}$$

$$\rho_3 := \min\left(\left|\{\zeta_i\}_{i=1}^{3}\right|\right) \tag{4.250}$$

respectively for each Ω_a with

$$1 \geq \rho_2 \geq \rho_1 \geq \rho_3 \geq 0 \tag{4.251}$$

where ρ_1 is defined as a lower principal root.

`Step 3:`

Plot ρ_2, ρ_1, and ρ_3 against $\log_{10}\Omega_a$.

Selected Algorithms (in Fig. 4.7):

U0V0/V0U0$(1.0, 1.0, 1.0)$: MPR-EPA (classical midpoint rule)

U0V0/V0U0$(0.8, 1.0, 0.8)$: U0V0/V0U0 Optimal

U0V0$(1.0, 1.0, 0.0)$: Implicit Newmark

V0U0$(1.0, 1.0, 0.0)$: MPR-MPA (new midpoint rule)

U0V1$(0.8, 0.8, 0.8)$: Three Parameter Optimal

V0U1$(0.8, 0.8, 0.8)$: V0 Counterpart and complements Three Parameter Optimal

U0V1$(0.0, 0.0, 0.0)$: L-Stable Three Parameter Optimal

V0U1$(0.0, 0.0, 0.0)$: V0 Counterpart and complements L-Stable Three Parameter Optimal

U0V1$(0.8, 0.8, 0.0)$: WBZ

V0U1$(0.8, 0.8, 0.0)$: V0 Counterpart and complements WBZ

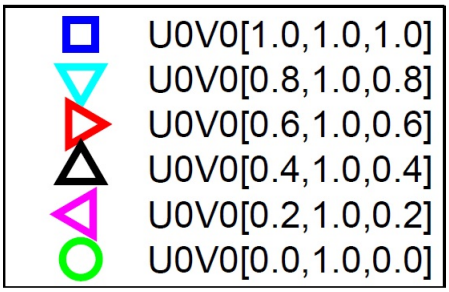
U0V1$(0.8, 0.9, 0.09)$: HHT-α ($\rho_\infty^{\mathrm{s}} = [1 - \rho_\infty^{\min}\rho_\infty^{\max}]/[\rho_\infty^{\min} + \rho_\infty^{\max} + 2\rho_\infty^{\min}\rho_\infty^{\max}]$)

V0U1$(0.8, 0.9, 0.09)$: V0 Counterpart and complements HHT-α

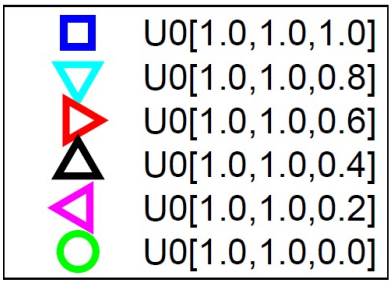
Mechanical Energy: Fig. 4.8-(a)(b) shows that the mechanical energy of the system is exactly conserved for any scheme obtained by selecting $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\mathrm{s}} \in [0, 1]$ in the U0 family or V0 family of algorithms in the linear dynamical system (as we predicted). But, this feature is not true in nonlinear dynamical

systems. We will carefully discuss this point later. Other plots in Fig. 4.8 show that the mechanical energy is dissipative for the numerically dissipative schemes. The enlarged legends of the figures are shown in Fig. 4.5.

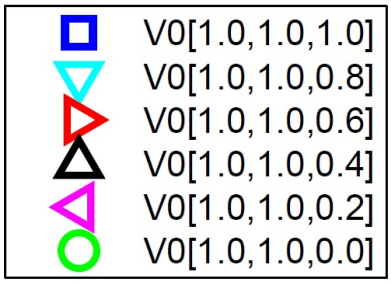
Numerical Dissipation and Numerical Dispersion: Fig. 4.9 (a)-(d) shows that there is no numerical dissipation for any scheme obtained by selecting $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\mathrm{s}} \in [0, 1]$ in the U0 family or V0 family of algorithms in the linear dynamical system. This is understandable since these schemes not only exactly conserve the mechanical energy of the system, but preserve the symplecticness in the conservative system. As we can see from the other plots in Fig. 4.9, there does not exist any scheme which has no numerical dispersion within the implicit family of the GSSSS algorithms. The enlarged legends of the figures are shown in Fig. 4.5.



(a) U0V0/V0U0(ρ_∞,1.0,ρ_∞)

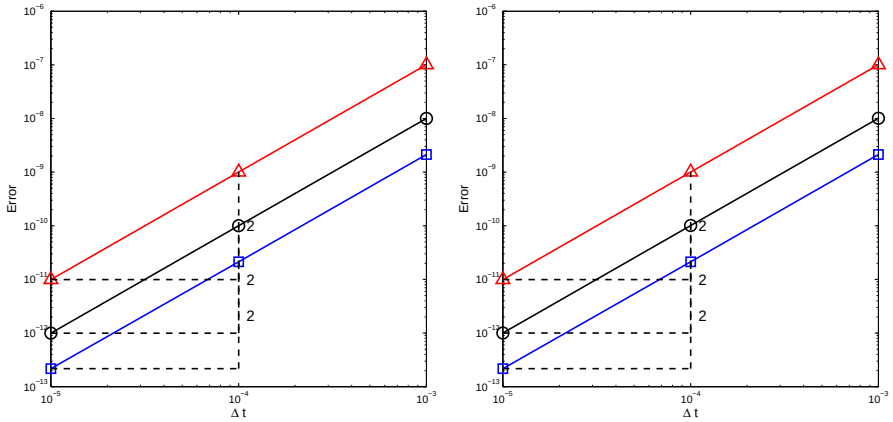


(b) U0V0(1.0,1.0,ρ^{s}_∞)



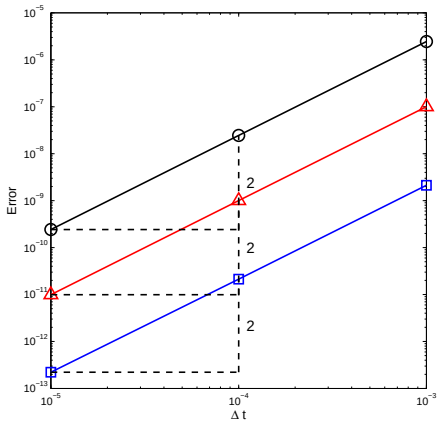
(c) V0U0(1.0,1.0,ρ^{s}_∞)

Figure 4.5: The summary of the legends of the figures

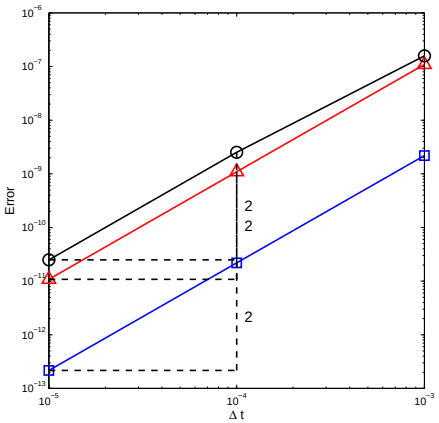


(a) U0V0/V0U0(1.0,1.0,1.0)

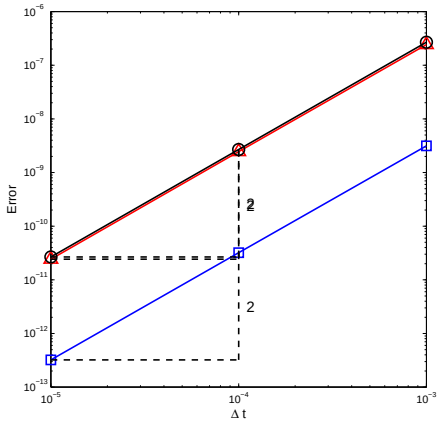
(b) U0V0(1.0,1.0,0.0)



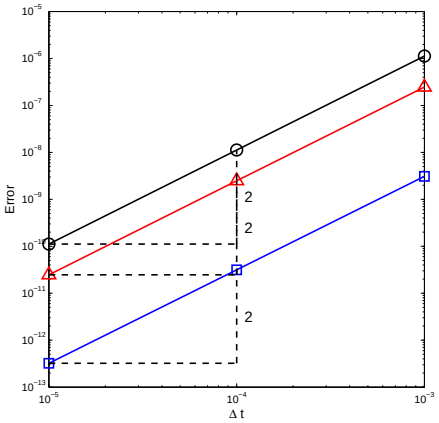
(c) V0U0(1.0,1.0,0.0)



(d) U0V0/V0U0(0.8,1.0,0.8)

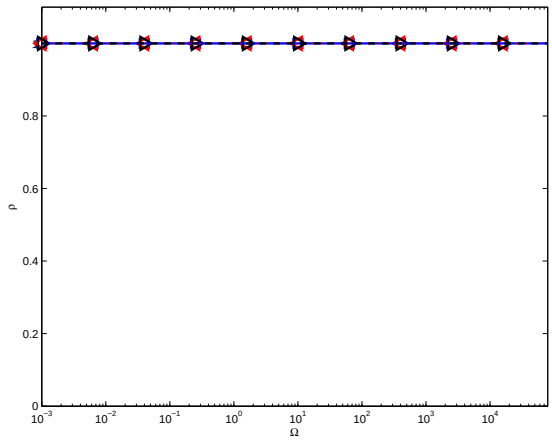


(e) U0V1(0.6,0.9,0.2)

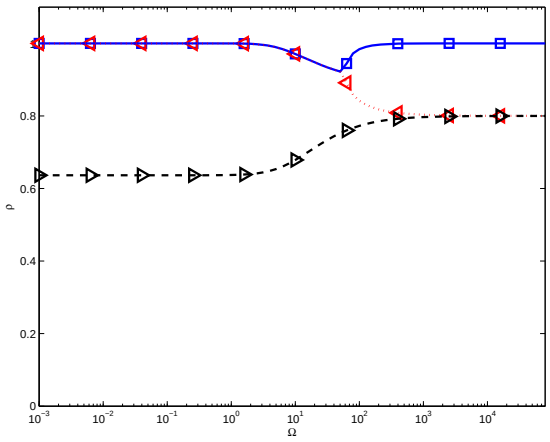


(f) V0U1(0.6,0.9,0.2)

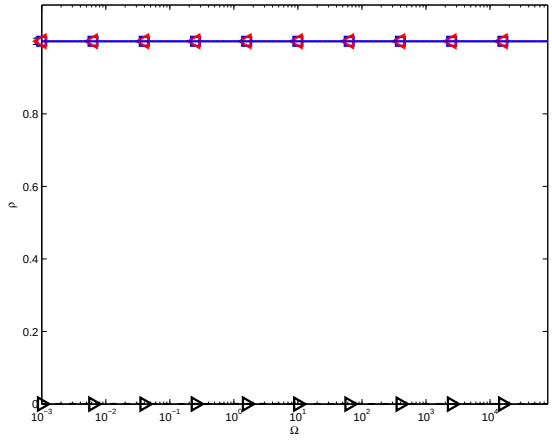
Figure 4.6: Time Accuracy plots in the configuration (□), velocity (△), and acceleration (○) within Algorithms 2 and 3 for the linear dissipative nonhomogeneous system, $\ddot{q} + \dot{q} + 10q = 0.5\sin(t)$, with the initial conditions $q_0 = 1$ and $\dot{q}_0 = 1$ - Single-field form



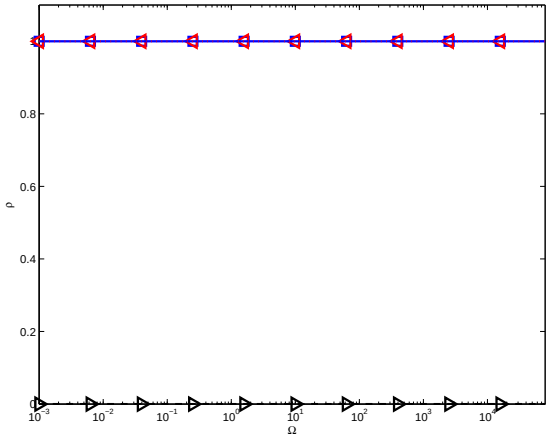
(a) U0V0/V0U0(1.0,1.0,1.0)



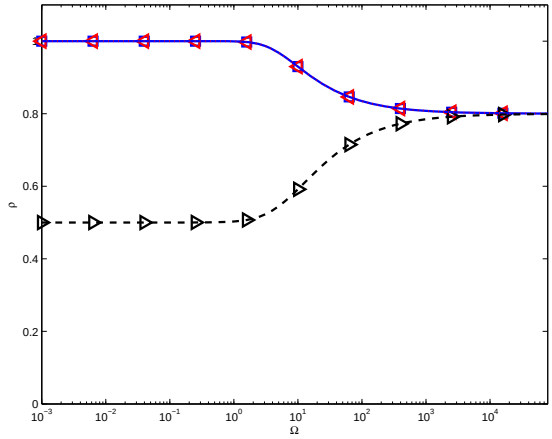
(b) U0V0/V0U0(0.8,1.0,0.8)



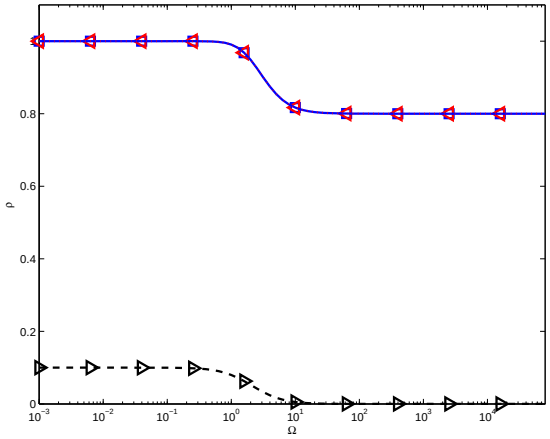
(c) U0V0(1.0,1.0,0.0)



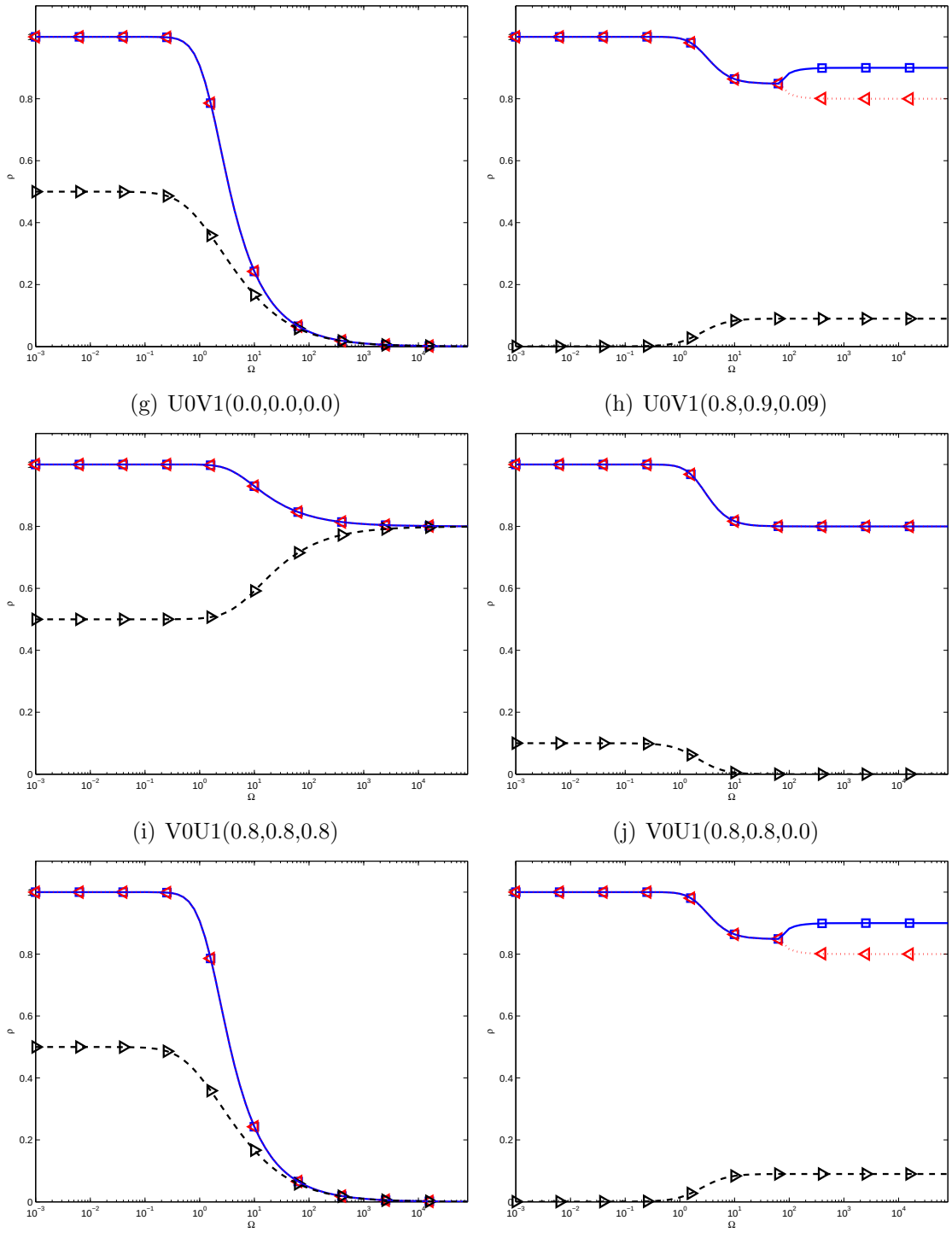
(d) V0U0(1.0,1.0,0.0)



(e) U0V1(0.8,0.8,0.8)



(f) U0V1(0.8,0.8,0.0)



(g) U0V1(0.0,0.0,0.0)

(h) U0V1(0.8,0.9,0.09)

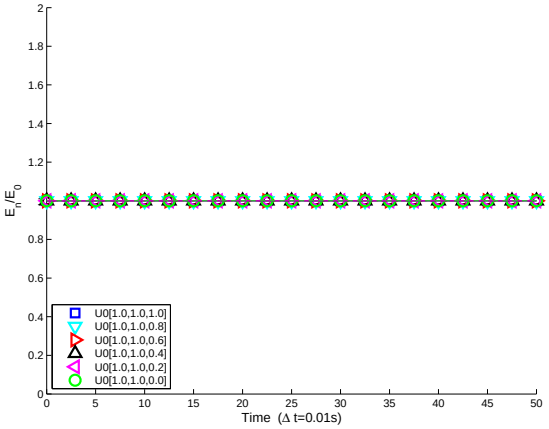
(i) V0U1(0.8,0.8,0.8)

(j) V0U1(0.8,0.8,0.0)

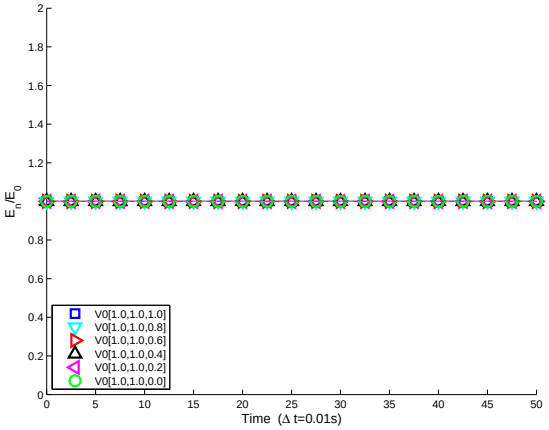
(k) V0U1(0.0,0.0,0.0)

(l) V0U1(0.8,0.9,0.09)

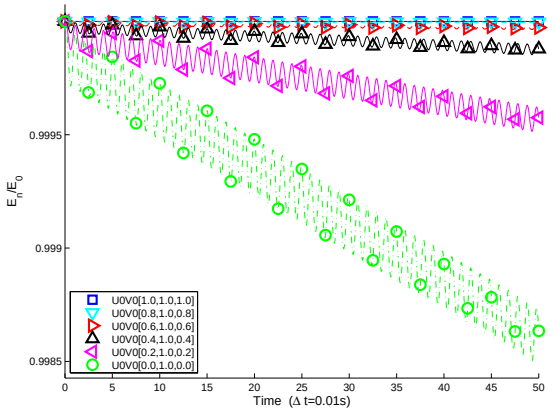
Figure 4.7: Stability plots of selected algorithms within Algorithms 2 and 3 in the conservative system, $\ddot{q} + q = 0$ ($\omega = 1$) - Single-field form



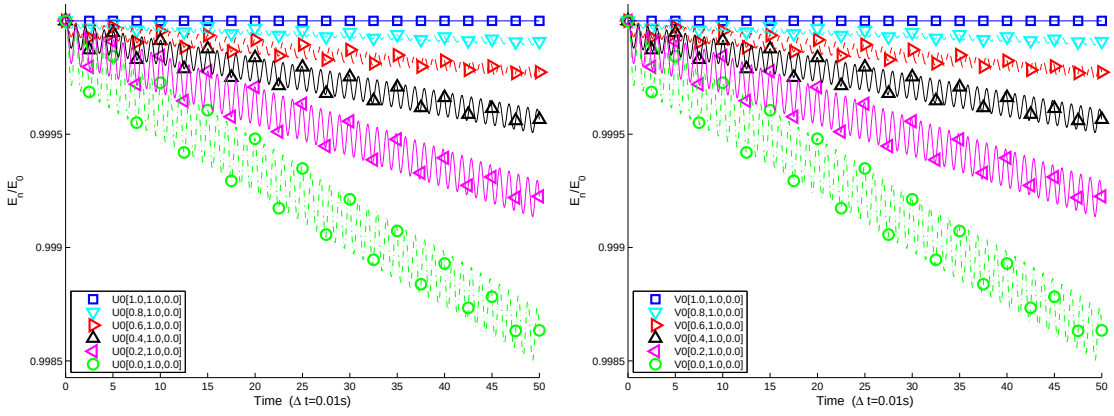
(a) U0V0(1,1,ρ^{s}_{∞})



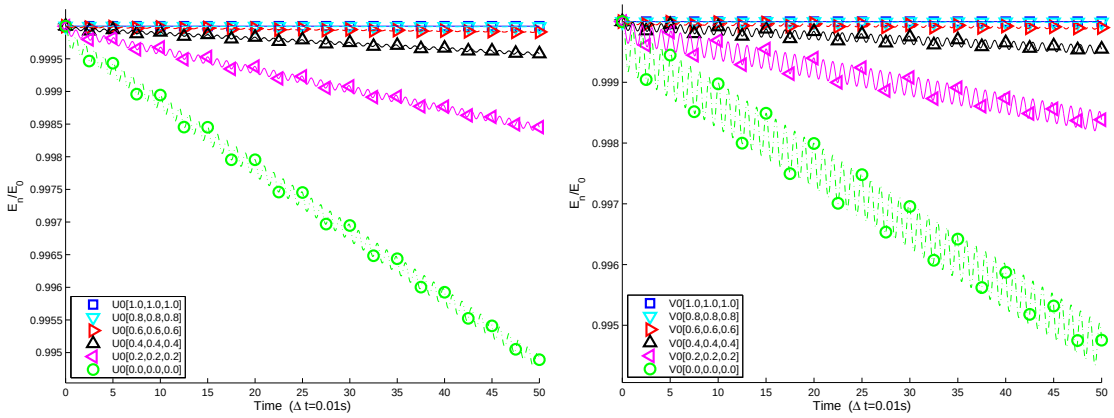
(b) V0U0(1,1,ρ^{s}_{∞})



(c) U0V0/V0U0 Optimal

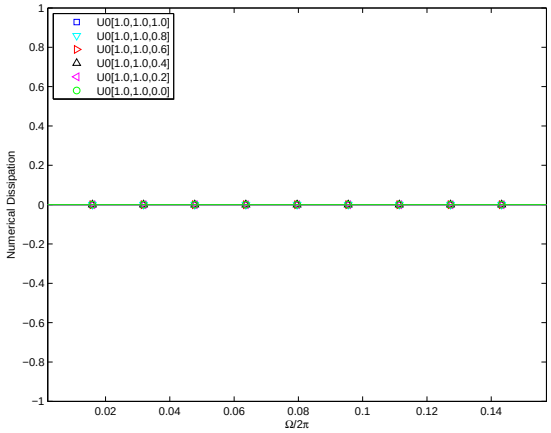


(d) U0V0($\rho_\infty^{\min}$,1,0) (e) V0U0($\rho_\infty^{\min}$,1,0)

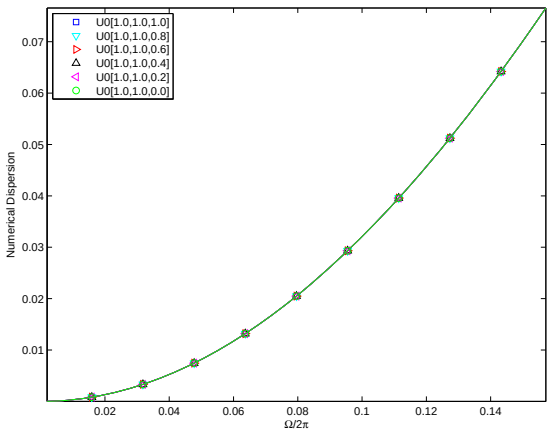


(f) U0V1(ρ_∞,ρ_∞,ρ_∞) (g) V0U1(ρ_∞,ρ_∞,ρ_∞)

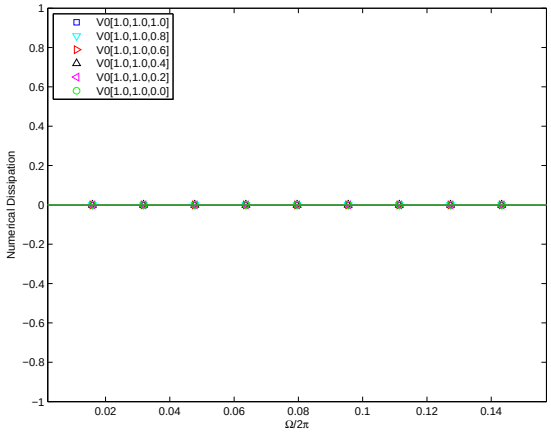
Figure 4.8: Mechanical Energy plots of the selected algorithms within Algorithm 2 and 3 ($\Delta t = 0.01$ sec) for the linear conservative homogeneous system, $\ddot{q}+10q = 0$, with the initial conditions $q_0 = 1$ and $\dot{q}_0 = 1$ - Single-field form



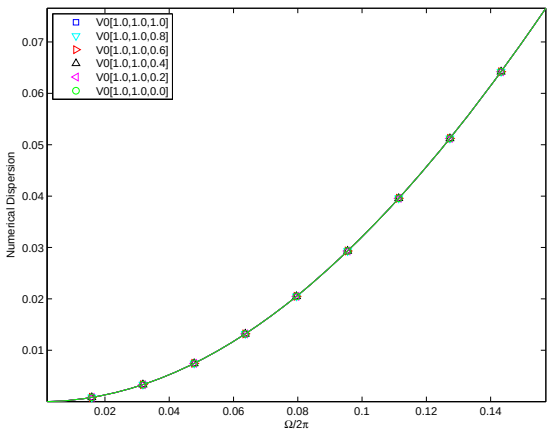
(a) U0V0(1,1,ρ^{s}_{∞}): Dissipation



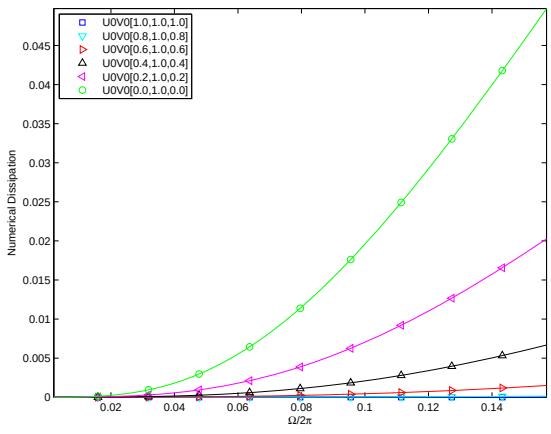
(b) U0V0(1,1,ρ^{s}_{∞}): Dispersion



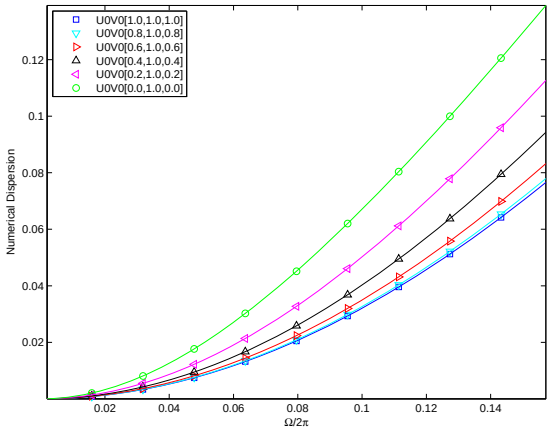
(c) V0U0(1,1,ρ^{s}_{∞}): Dissipation



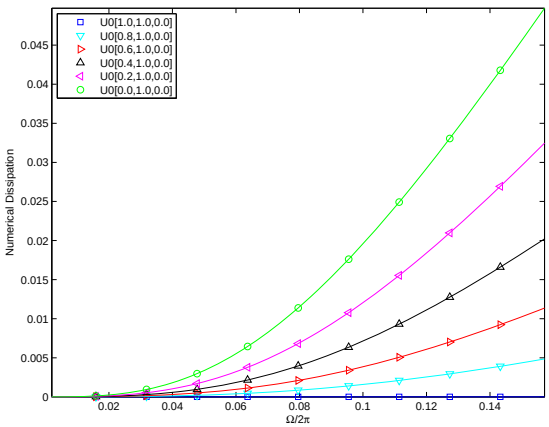
(d) V0U0(1,1,ρ^{s}_{∞}): Dispersion



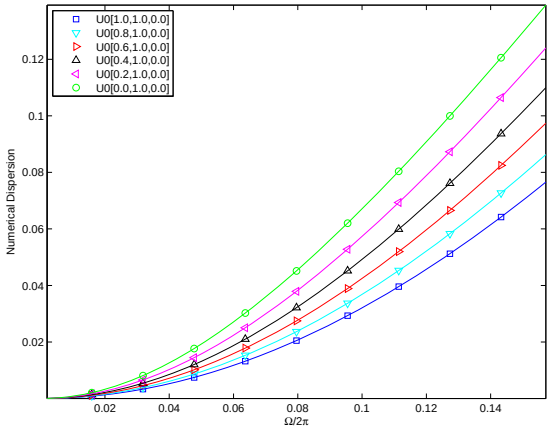
(e) U0V0/V0U0 Optimal: Dissipation



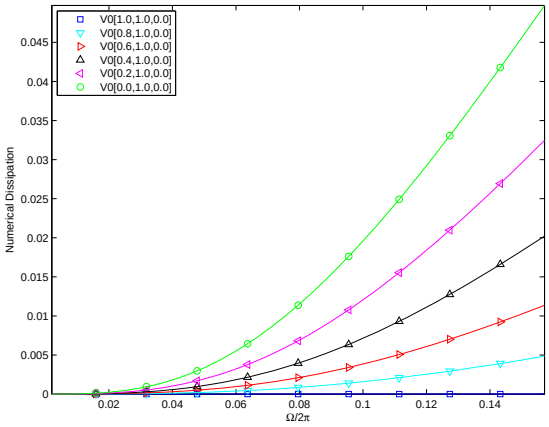
(f) U0V0/V0U0 Optimal: Dispersion



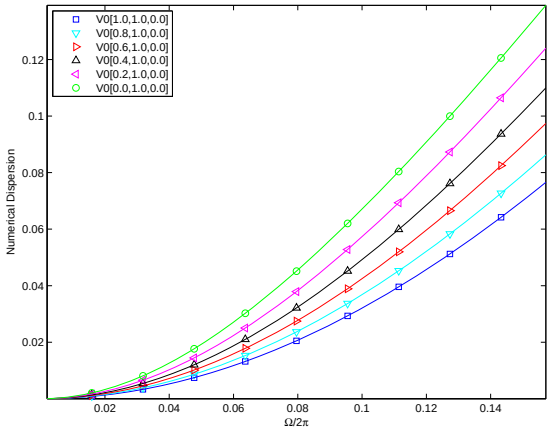
(g) U0V0($\rho_\infty^{\min}$,1,0): Dissipation



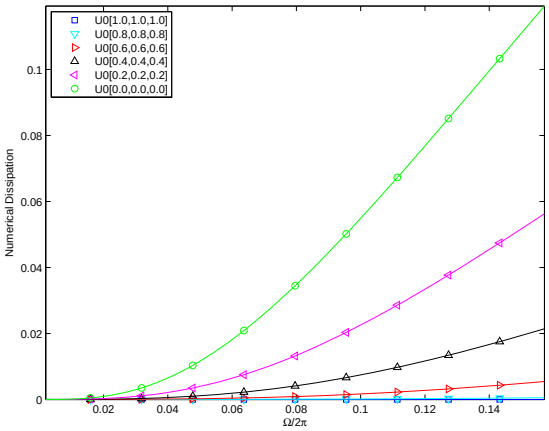
(h) U0V0($\rho_\infty^{\min}$,1,0): Dispersion



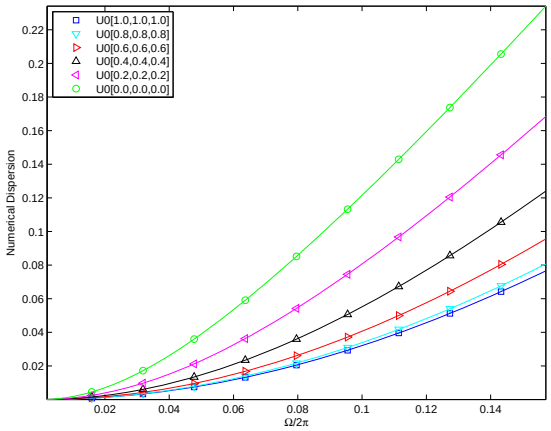
(i) V0U0($\rho_\infty^{\min}$,1,0): Dissipation



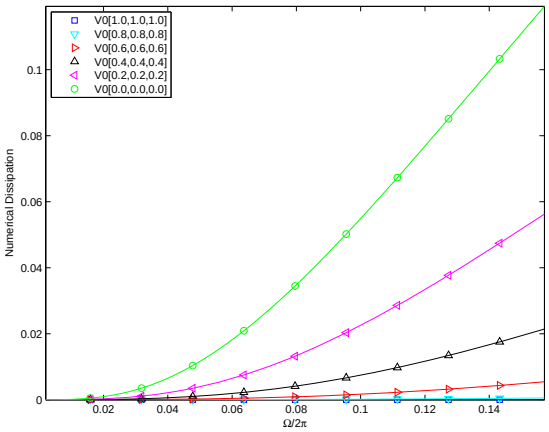
(j) V0U0($\rho_\infty^{\min}$,1,0): Dispersion



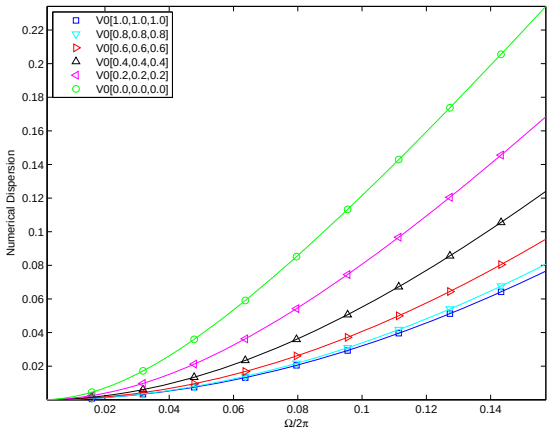
(k) U0V1(ρ_∞,ρ_∞,ρ_∞): Dissipation



(l) U0V1(ρ_∞,ρ_∞,ρ_∞): Dispersion



(m) V0U1(ρ_∞,ρ_∞,ρ_∞): Dissipation



(n) V0U1(ρ_∞,ρ_∞,ρ_∞): Dispersion

Figure 4.9: Numerical dissipation and numerical dispersion plots of selected algorithms within Algorithms 2 and 3 in the conservative system, $\ddot{q} + q = 0$ ($\omega = 1$) - Single-field form

Chapter 5

Explicit Generalized Single Step Single Solve Framework and Family of Algorithms in Two- and Single-Field Forms - A Unified Framework: Linear Dynamical Systems

In this chapter, we derive the family of the explicit generalized single step single solve algorithms in the two- and single-field forms for linear dynamical systems. It consists of the two groups; that is, (i) the family of the predictor-corrector explicit GSSSS algorithms (PCE-GSSSS family of algorithms), which is directly derived from the family of implicit generalized single step single solve (I-GSSSS) algorithms, and (ii) the family of the (general) explicit GSSSS algorithms (E-GSSSS family of algorithms). In contrast to the PCE-GSSSS family of algorithms, the E-GSSSS family of algorithms are not derived from the implicit GSSSS family of algorithms. Here, we relax the conditions for the unconditional stability that

we first imposed and then employed in deriving the general structure of I-GSSSS framework from which we designed the PCE-GSSSS framework; and we instead independently design the general structure and form of the single step single solve family of general explicit algorithms. These two explicit frameworks and the implicit framework will be subsequently unified into a single algorithmic architecture in the single-field form at the end.

5.1 PCE-GSSSS Framework of Algorithms in Two-Field Form

Consider the equation of motion for the linearized equation of motion in two-field form with the kinematic constraint and the given initial conditions; see Eq. (4.1). In Chapter 4, the two-field form I-GSSSS framework of algorithms for linear dynamical systems, i.e., Algorithm 1, was designed for the system. To design the two-field form PCE-GSSSS framework of algorithms, we consider the following two cases for a damped system: (i) implicit treatment of the velocity term, and (ii) explicit treatment of the velocity term.

5.1.1 Implicit Treatment of the Velocity Term

Algorithm 1 shows that the algorithmic $\mathbf{q}(t)$ and $\boldsymbol{\nu}(t)$ take the form

$$\tilde{\mathbf{q}} = \mathbf{q}_n + \frac{W_2\Lambda_2}{\lambda_2}(\mathbf{q}_{n+1} - \mathbf{q}_n) + \left(W_1\Lambda_1 - \frac{\lambda_1}{\lambda_2}W_2\Lambda_2\right)\boldsymbol{\nu}_n\Delta t \tag{5.1}$$

$$\tilde{\boldsymbol{\nu}} = \boldsymbol{\nu}_n + W_1\Lambda_4(\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n) \tag{5.2}$$

respectively. If we treat the dissipative force term implicitly, the two-field form PCE-GSSSS framework of algorithms can be designed as follows:

Predictor:

$$\hat{\mathbf{q}} = \mathbf{q}_n + (\lambda_1 - \lambda_2)\boldsymbol{\nu}_n\Delta t \tag{5.3}$$

Balance Equation:

$$\mathbf{M}\frac{\boldsymbol{\nu}_{n+1}-\boldsymbol{\nu}_n}{\Delta t}+\mathbf{C}\tilde{\boldsymbol{\nu}}+\mathbf{K}\breve{\mathbf{q}}=\tilde{\mathbf{f}} \tag{5.4}$$

where

$$\begin{aligned}
\breve{\mathbf{q}} &= \mathbf{q}_n+\frac{W_2\Lambda_2}{\lambda_2}(\hat{\mathbf{q}}-\mathbf{q}_n)+\left(W_1\Lambda_1-\frac{\lambda_1}{\lambda_2}W_2\Lambda_2\right)\boldsymbol{\nu}_n\Delta t && (5.5)\\
\tilde{\boldsymbol{\nu}} &= (1-W_1\Lambda_4)\boldsymbol{\nu}_n+W_1\Lambda_4\boldsymbol{\nu}_{n+1} && (5.6)\\
\tilde{\mathbf{f}} &= (1-W_1)\mathbf{f}_n+W_1\mathbf{f}_{n+1} && (5.7)
\end{aligned}$$

Corrector:

$$\mathbf{q}_{n+1} = \hat{\mathbf{q}}+\lambda_2\boldsymbol{\nu}_{n+1}\Delta t \tag{5.8}$$

Initial conditions:

$$\begin{aligned}
\mathbf{q}(t_0) &= \mathbf{q}_0\\
\boldsymbol{\nu}(t_0) &= \boldsymbol{\nu}_0
\end{aligned} \tag{5.9}$$

Note that the algorithmic external force is given as $\tilde{\mathbf{f}}=\mathbf{f}_{n+W_1}$ or $\tilde{\mathbf{f}}=\mathbf{f}(t_{n+W_1})$. The single-degree-of-freedom system representation by mode superposition is readily obtained as follows:

Predictor:

$$\hat{q}=q_n+(\lambda_1-\lambda_2)\nu_n\Delta t \tag{5.10}$$

Balance Equation:

$$\frac{\nu_{n+1}-\nu_n}{\Delta t}+2\xi\omega\tilde{\nu}+\omega^2\breve{q}=\tilde{g} \tag{5.11}$$

where

$$\begin{aligned}
\breve{q} &= q_n+\frac{W_2\Lambda_2}{\lambda_2}(\hat{q}-q_n)+\left(W_1\Lambda_1-\frac{\lambda_1}{\lambda_2}W_2\Lambda_2\right)\nu_n\Delta t && (5.12)\\
\tilde{\nu} &= (1-W_1\Lambda_4)\nu_n+W_1\Lambda_4\nu_{n+1} && (5.13)\\
\tilde{g} &= (1-W_1)g_n+W_1g_{n+1} && (5.14)
\end{aligned}$$

Corrector:

$$q_{n+1} = \hat{q}+\lambda_2\nu_{n+1}\Delta t \tag{5.15}$$

Initial conditions:

$$\begin{aligned} q(t_0) &= q_0 \\ \nu(t_0) &= \nu_0 \end{aligned} \tag{5.16}$$

Note that ν_{n+1} is obtained from Eq. (5.11) as

$$\nu_{n+1} = \frac{(\tilde{g} - \omega^2 \breve{q})\Delta t + [1 - 2\xi\omega(1 - W_1\Lambda_4)\Delta t]\,\nu_n}{1 + 2\xi\omega W_1\Lambda_4\Delta t} \tag{5.17}$$

And, Eq. (5.10) - Eq. (5.17) can be cast into the following expression:

$$\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n + \mathbf{L}_{n+W_1} \tag{5.18}$$

where the amplification matrix and the load vector can be written as

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} 1 + \lambda_2\alpha_1 & \lambda_1 + \lambda_2\alpha_2 \\ \alpha_1 & 1 + \alpha_2 \end{bmatrix} \\ \mathbf{L}_{n+W_1} &= \frac{\Delta t^2}{D}\left[(1 - W_1)g_n + W_1 g_{n+1}\right] \begin{pmatrix} \lambda_2 \\ 1 \end{pmatrix} \end{aligned} \tag{5.19}$$

in which

$$\begin{aligned} &\alpha_1 = -\frac{\Omega^2}{D}, \quad \alpha_2 = -\frac{2\xi\Omega + (W_1\Lambda_1 - W_2\Lambda_2)\Omega^2}{D} \\ &D = 1 + 2\xi\Omega W_1\Lambda_4 \\ &\Omega = \omega\Delta t \end{aligned} \tag{5.20}$$

and $\mathbf{y}_{n+1} = \{q_{n+1}, \Delta t\nu_{n+1}\}^T$ and $\mathbf{y}_n = \{q_n, \Delta t\nu_n\}^T$. The principal invariants of $\mathbf{A} \in \mathbb{R}^{2\times 2}$ are given as

$$\begin{aligned} A_1 &= \mathrm{tr}(\mathbf{A}) \\ &= \frac{2 - 2(1 - 2W_1\Lambda_4)\xi\Omega - (W_1\Lambda_1 + \lambda_2 - W_2\Lambda_2)\Omega^2}{D} \end{aligned} \tag{5.21}$$

$$\begin{aligned} A_2 &= \det(\mathbf{A}) \\ &= \frac{1 - 2(1 - W_1\Lambda_4)\xi\Omega + (\lambda_1 - W_1\Lambda_1 - \lambda_2 + W_2\Lambda_2)\Omega^2}{D} \end{aligned} \tag{5.22}$$

Order of Time Accuracy

Again, consider the homogeneous case for simplicity. The difference equation is obtained as

$$\mathbf{y}_{n+2} - A_1\mathbf{y}_{n+1} + A_2\mathbf{y}_n - A_3\mathbf{y}_{n-1} = \mathbf{0} \tag{5.23}$$

We define the local truncation error for q in the sense of the linear-multistep (LMS) method as

$$\tau_q^{(1)}(\Delta t) := \frac{1}{\Delta t^2}\left[q(t_{n+2}) - A_1 q(t_{n+1}) + A_2 q(t_n)\right] \tag{5.24}$$

where $q(t_{n+2})$, $q(t_{n+1})$, and $q(t_n)$ denote the exact solutions at time t_{n+2}, t_{n+1}, and t_n, respectively, i.e., $q(t_{n+2}) \approx q_{n+2}$, $q(t_{n+1}) \approx q_{n+1}$, and $q(t_n) \approx q_n$. Substituting the Taylor series expansions of $q(t_{n+2})$ and $q(t_{n+1})$ at time t_n, i.e., Eq. (4.38) and Eq. (4.39), respectively, into Eq. (5.24) yields

$$\begin{aligned}\tau_q^{(1)}(\Delta t) = {} & \frac{1}{D}\left[\ddot{q}(t_n) + 2\omega\xi\dot{q}(t_n) + \lambda_1\omega^2 q(t_n)\right] \\ & + \frac{\Delta t}{D}\left[\dddot{q}(t_n) + (1 + 2W_1\Lambda_4)\omega\xi\ddot{q}(t_n) + (\lambda_2 + W_1\Lambda_1 - W_2\Lambda_2)\omega^2\dot{q}(t_n)\right] \\ & + \mathcal{O}(\Delta t^2)\end{aligned} \tag{5.25}$$

or

$$\begin{aligned}\tau_q^{(1)}(\Delta t) = {} & \frac{1}{D}\left[\dot{\nu}(t_n) + 2\omega\xi\nu(t_n) + \lambda_1\omega^2 q(t_n)\right] \\ & + \frac{\Delta t}{D}\left[\ddot{\nu}(t_n) + (1 + 2W_1\Lambda_4)\omega\xi\dot{\nu}(t_n) + (\lambda_2 + W_1\Lambda_1 - W_2\Lambda_2)\omega^2\nu(t_n)\right] \\ & + \mathcal{O}(\Delta t^2)\end{aligned} \tag{5.26}$$

due to the kinematic constraint. Hence, $\tau_q^{(1)}(\Delta t) = \mathcal{O}(\Delta t)$ is obtained if $\lambda_1 = 1$ and all other algorithmic parameters are arbitrary. However, $\tau_q^{(1)}(\Delta t) = \mathcal{O}(\Delta t^2)$ is obtained if

$$\begin{aligned}\lambda_1 &= 1 \\ W_1\Lambda_4 &= \frac{1}{2} \\ \lambda_2 + W_1\Lambda_1 - W_2\Lambda_2 &= 1\end{aligned} \tag{5.27}$$

The above conditions given by Eq. (5.27) are not the necessary and sufficient conditions. We define the local truncation error for q from $\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n$ as

$$\tau_q^{(2)}(\Delta t) := q(t_{n+1}) - A_{11}q(t_n) - \Delta t A_{12}\nu(t_n) \tag{5.28}$$

Substituting the Taylor series expansion of $q(t_{n+1})$ at time t_n, i.e., Eq. (4.39), we get

$$\begin{aligned}\tau_q^{(2)}(\Delta t) = \frac{\Delta t}{D}(1-\lambda_1)\nu(t_n) \\ + \frac{\Delta t^2}{2D}[\dot{\nu}(t_n) + 4\left(\lambda_2 + (1-\lambda_1)W_1\Lambda_4\right)\xi\omega\nu(t_n) \\ + 2\lambda_2\omega^2 q(t_n)] + \mathcal{O}(\Delta t^3)\end{aligned} \tag{5.29}$$

Therefore, $\tau_q^{(2)}(\Delta t) = \mathcal{O}(\Delta t)$ is obtained for arbitrary algorithmic parameters; however, $\tau_q^{(2)}(\Delta t) = \mathcal{O}(\Delta t^2)$ is obtained if $\lambda_1 = 1$ and all other algorithmic parameters are arbitrary; and $\tau_q^{(2)}(\Delta t) = \mathcal{O}(\Delta t^3)$ is obtained if

$$\lambda_1 = 1 \;\text{ and }\; \lambda_2 = \frac{1}{2} \tag{5.30}$$

and all other algorithmic parameters are arbitrary. To guarantee the second-order time accuracy of algorithms, we must have both $\tau_q^{(1)}(\Delta t) = \mathcal{O}(\Delta t^2)$ and $\tau_q^{(2)}(\Delta t) = \mathcal{O}(\Delta t^3)$. Hence, the necessary and sufficient condition for the second-order time accuracy of an algorithm is:

$$\boxed{\begin{aligned}\lambda_1 = 1,\;\; \lambda_2 = \frac{1}{2},\;\; W_1\Lambda_4 = \frac{1}{2} \\ W_1\Lambda_1 - W_2\Lambda_2 = \frac{1}{2} \\ \textbf{(for homogeneous case)}\end{aligned}} \tag{5.31}$$

The above conditions can actually be obtained without using $\tau_q^{(1)}(\Delta t)$ as defined in Eq. (5.24). We define the local truncation error for ν from $\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n$ as

$$\tau_\nu^{(2)}(\Delta t) := \nu(t_{n+1}) - \frac{A_{21}}{\Delta t}q(t_n) - A_{22}\nu(t_n) \tag{5.32}$$

Substituting the Taylor series expansion of $\nu(t_{n+1})$ at time t_n,

$$\nu(t_{n+1}) = \nu(t_n) + \Delta t\dot{\nu}(t_n) + \frac{\Delta t^2}{2}\ddot{\nu}(t_n) + \mathcal{O}(\Delta t^3) \tag{5.33}$$

we get

$$\tau_\nu^{(2)}(\Delta t) = \frac{\Delta t^2}{2D}\left[\dot{\nu}(t_n) + 4W_1\Lambda_4\xi\omega\nu(t_n) + 2(W_1\Lambda_1 - W_2\Lambda_2)\omega^2\boldsymbol{\nu}(t_n)\right] + \mathcal{O}(\Delta t^3) \tag{5.34}$$

Therefore, $\tau_\nu^{(2)}(\Delta t) = \mathcal{O}(\Delta t^2)$ is obtained for arbitrary algorithmic parameters; and $\tau_\nu^{(2)}(\Delta t) = \mathcal{O}(\Delta t^3)$ is obtained if

$$W_1\Lambda_1 - W_2\Lambda_2 = \frac{1}{2} \text{ and } W_1\Lambda_4 = \frac{1}{2} \tag{5.35}$$

and all other algorithmic parameters are arbitrary. Hence, the necessary and sufficient conditions as shown in Eq. (5.31) can be also obtained from $\tau_q^{(2)}(\Delta t) = (\Delta t^3)$ and $\tau_\nu^{(2)}(\Delta t) = (\Delta t^3)$.

For the non-homogeneous case, the necessary and sufficient conditions for the second-order time accuracy of an algorithm re:

$$\boxed{\begin{aligned}\lambda_1 = 1,\ \ \lambda_2 = \frac{1}{2},\ \ \Lambda_4 = 1,\ \ W_1 = \frac{1}{2}\\ \text{and } W_2\Lambda_2 = \frac{\Lambda_1 - 1}{2}\\ \textbf{(for non-homogeneous case)}\end{aligned}} \tag{5.36}$$

The derivation is straightforward and is hence omitted.

Spectral Analysis

In the PCE-GSSSS framework of algorithms, we employ the same algorithmic parameters from the I-GSSSS framework of algorithms. Recall that the second-order time accurate two-field form I-GSSSS framework of algorithms, i.e., Algorithm 1, has the algorithmic parameters,

$$\begin{aligned}\lambda_1 = \Lambda_1 = 1,\ \ \lambda_2 = \Lambda_2 = \frac{1}{2},\ \ \Lambda_4 = 1\\ W_1 = W_2 = \frac{1}{2}\end{aligned} \tag{5.37}$$

for the non-homogeneous case. Comparing Eq. (5.36) and Eq. (5.37), we clearly see that there exists no second-order time accurate two-field form PCE-GSSSS framework of algorithms unless otherwise we set

$$\boxed{\Lambda_2 \equiv 0} \tag{5.38}$$

in the PCE-GSSSS framework of algorithms. However, no $\rho_\infty^{\min}$ and $\rho_\infty^{\max}$ in $[0,1]$ can satisfy the following condition

$$W_2\Lambda_2 = \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} = 0 \tag{5.39}$$

Note that $W_1 = 1/2$ restricts the spectral condition to be $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ (see Eq. (4.69)), which is clearly in contradiction with Eq. (5.39)! Therefore, we conclude that ***there exists no second-order time accurate PCE-GSSSS framework of algorithms in the two-field form with the implicit treatment of the velocity term.*** In the two-field form I-GSSSS framework of algorithms design, the algorithmic parameter $W_2\Lambda_2$ is arbitrary so as to guarantee second-order time accuracy; but it has been related to the absolute values of the eigenvalues of the amplification matrix for the implicit case at the high-frequency range, i.e., $\rho_\infty^{\min}, \rho_\infty^{\max} \in [0,1]$, as shown in Eq. (4.69), for the unconditional stability requirement. However, note that for the case of the associated PCE-GSSSS design, $\rho_\infty^{\min}$ and $\rho_\infty^{\max}$ can be no longer considered (not meaningful) as the spectral radius at the high-frequency range since we have conditionally stable or unstable schemes.

5.1.2 Explicit Treatment of the Velocity Term

In contrast, if we treat the dissipative force term explicitly, the two-field form PCE-GSSSS framework of algorithms is obtained as follows:

Predictors:

$$\hat{\mathbf{q}} = \mathbf{q}_n + (\lambda_1 - \lambda_2)\boldsymbol{\nu}_n \Delta t \tag{5.40}$$

$$\hat{\boldsymbol{\nu}} = \mathbf{0} \tag{5.41}$$

Balance Equation:

$$\mathbf{M}\frac{\boldsymbol{\nu}_{n+1}-\boldsymbol{\nu}_n}{\Delta t}+\mathbf{C}\breve{\boldsymbol{\nu}}+\mathbf{K}\breve{\mathbf{q}}=\tilde{\mathbf{f}} \tag{5.42}$$

where

$$\begin{aligned}
\breve{\mathbf{q}} &= \mathbf{q}_n+\frac{W_2\Lambda_2}{\lambda_2}(\hat{\mathbf{q}}-\mathbf{q}_n)+\left(W_1\Lambda_1-\frac{\lambda_1}{\lambda_2}W_2\Lambda_2\right)\boldsymbol{\nu}_n\Delta t && (5.43)\\
\breve{\boldsymbol{\nu}} &= (1-W_1\Lambda_4)\boldsymbol{\nu}_n+W_1\Lambda_4\hat{\boldsymbol{\nu}}=(1-W_1\Lambda_4)\boldsymbol{\nu}_n && (5.44)\\
\tilde{\mathbf{f}} &= (1-W_1)\mathbf{f}_n+W_1\mathbf{f}_{n+1} && (5.45)
\end{aligned}$$

Correctors:

$$\begin{aligned}
\mathbf{q}_{n+1} &= \hat{\mathbf{q}}+\lambda_2\boldsymbol{\nu}_{n+1}\Delta t && (5.46)\\
\boldsymbol{\nu}_{n+1} &= \hat{\boldsymbol{\nu}}+\boldsymbol{\nu}_{n+1} && (5.47)
\end{aligned}$$

Initial conditions:

$$\begin{aligned}
\mathbf{q}(t_0)&=\mathbf{q}_0\\
\boldsymbol{\nu}(t_0)&=\boldsymbol{\nu}_0
\end{aligned} \tag{5.48}$$

Note that the algorithmic external force is given as $\tilde{\mathbf{f}}=\mathbf{f}_{n+W_1}$ or $\tilde{\mathbf{f}}=\mathbf{f}(t_{n+W_1})$. The single-degree-of-freedom system representation by mode superposition is readily obtained as follows:

Predictors:

$$\begin{aligned}
\hat{q} &= q_n+(\lambda_1-\lambda_2)\nu_n\Delta t && (5.49)\\
\hat{\nu} &= 0 && (5.50)
\end{aligned}$$

Balance Equation:

$$\frac{\nu_{n+1}-\nu_n}{\Delta t}+2\xi\omega\breve{\nu}+\omega^2\breve{q}=\tilde{g} \tag{5.51}$$

where

$$\begin{aligned}
\breve{q} &= q_n+\frac{W_2\Lambda_2}{\lambda_2}(\hat{q}-q_n)+\left(W_1\Lambda_1-\frac{\lambda_1}{\lambda_2}W_2\Lambda_2\right)\nu_n\Delta t && (5.52)\\
\breve{\nu} &= (1-W_1\Lambda_4)\nu_n+W_1\Lambda_4\hat{\nu}=(1-W_1\Lambda_4)\nu_n && (5.53)\\
\tilde{g} &= (1-W_1)g_n+W_1g_{n+1} && (5.54)
\end{aligned}$$

Correctors:

$$\begin{aligned} q_{n+1} &= \hat{q} + \lambda_2 \nu_{n+1} \Delta t \qquad (5.55) \\ \nu_{n+1} &= \hat{\nu} + \nu_{n+1} \qquad (5.56) \end{aligned}$$

Initial conditions:

$$\begin{aligned} q(t_0) &= q_0 \\ \nu(t_0) &= \nu_0 \end{aligned} \qquad (5.57)$$

Note that ν_{n+1} is obtained from Eq. (5.51) as

$$\nu_{n+1} = \nu_n + \left[\tilde{g} - \omega^2 \breve{q} - 2\xi\omega\breve{\nu}\right] \Delta t \qquad (5.58)$$

And, Eq. (5.49) - Eq. (5.58) can be cast into the following expression:

$$\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n + \mathbf{L}_{n+W_1} \qquad (5.59)$$

where the amplification matrix and the load vector can be written as

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} 1 + \lambda_2\alpha_1 & \lambda_1 + \lambda_2\alpha_2 \\ \alpha_1 & 1 + \alpha_2 \end{bmatrix} \\ \mathbf{L}_{n+W_1} &= \Delta t^2 \left[(1 - W_1) g_n + W_1 g_{n+1}\right] \begin{pmatrix} \lambda_2 \\ 1 \end{pmatrix} \end{aligned} \qquad (5.60)$$

in which

$$\begin{aligned} \alpha_1 &= -\Omega^2 \\ \alpha_2 &= -2(1 - W_1\Lambda_4)\xi\Omega - (W_1\Lambda_1 - W_2\Lambda_2)\Omega^2 \\ \Omega &= \omega\Delta t \end{aligned} \qquad (5.61)$$

and $\mathbf{y}_{n+1} = \{q_{n+1}, \Delta t \nu_{n+1}\}^T$ and $\mathbf{y}_n = \{q_n, \Delta t \nu_n\}^T$. The principal invariants of $\mathbf{A} \in \mathbb{R}^{2\times 2}$ are given as

$$\begin{aligned} A_1 &= \mathrm{tr}(\mathbf{A}) \\ &= 2 - 2(1 - W_1\Lambda_4)\xi\Omega - (W_1\Lambda_1 + \lambda_2 - W_2\Lambda_2)\Omega^2 \qquad (5.62) \\ A_2 &= \det(\mathbf{A}) \\ &= 1 - 2(1 - W_1\Lambda_4)\xi\Omega + (\lambda_1 - W_1\Lambda_1 - \lambda_2 + W_2\Lambda_2)\Omega^2 \qquad (5.63) \end{aligned}$$

Order of Time Accuracy

Again, consider the homogeneous case for simplicity. Again, consider the homogeneous case for simplicity. The local truncation error for q in the sense of the linear-multistep (LMS) method is defined from the difference equation, $\mathbf{y}_{n+2} - A_1\mathbf{y}_{n+1} + A_2\mathbf{y}_n = \mathbf{0}$, as

$$\tau_q^{(1)}(\Delta t) := \frac{1}{\Delta t^2}\left[q(t_{n+2}) - A_1 q(t_{n+1}) + A_2 q(t_n)\right] \tag{5.64}$$

where $q(t_{n+2})$, $q(t_{n+1})$, and $q(t_n)$ denote the exact solutions at time t_{n+2}, t_{n+1}, and t_n, respectively, i.e., $q(t_{n+2}) \approx q_{n+2}$, $q(t_{n+1}) \approx q_{n+1}$, and $q(t_n) \approx q_n$. Substituting the Taylor series expansions of $q(t_{n+2})$ and $q(t_{n+1})$ at time t_n, i.e., Eq. (4.38) and Eq. (4.39), respectively, into Eq. (5.24) yields

$$\begin{aligned}\tau_q^{(1)}(\Delta t) = &\left[\ddot{q}(t_n) + 2(1 - W_1\Lambda_4)\omega\xi\dot{q}(t_n) + \lambda_1\omega^2 q(t_n)\right] \\ &+ \Delta t\left[\dddot{q}(t_n) + (1 - W_1\Lambda_4)\omega\xi\ddot{q}(t_n) + (\lambda_2 + W_1\Lambda_1 - W_2\Lambda_2)\omega^2\dot{q}(t_n)\right] \\ &+ \mathcal{O}(\Delta t^2)\end{aligned} \tag{5.65}$$

or

$$\begin{aligned}\tau_q^{(1)}(\Delta t) = &\left[\dot{\nu}(t_n) + 2(1 - W_1\Lambda_4)\omega\xi\nu(t_n) + \lambda_1\omega^2 q(t_n)\right] \\ &+ \Delta t\left[\ddot{\nu}(t_n) + (1 - W_1\Lambda_4)\omega\xi\dot{\nu}(t_n) + (\lambda_2 + W_1\Lambda_1 - W_2\Lambda_2)\omega^2\nu(t_n)\right] \\ &+ \mathcal{O}(\Delta t^2)\end{aligned} \tag{5.66}$$

due to the kinematic constraint. Hence, $\tau_q^{(1)}(\Delta t) = \mathcal{O}(\Delta t)$ is obtained if $\lambda_1 = 1 - W_1\Lambda_4 = 1$, i.e., $\lambda_1 = 1$ and $W_1\Lambda_4 = 0$, and all other algorithmic parameters are arbitrary. And $\tau_q^{(1)}(\Delta t) = \mathcal{O}(\Delta t^2)$ is obtained if

$$\begin{aligned}\lambda_1 &= 1 \\ 1 - W_1\Lambda_4 &= 1 \\ 1 - W_1\Lambda_4 &= \frac{1}{2} \\ \lambda_2 + W_1\Lambda_1 - W_2\Lambda_2 &= 1\end{aligned} \tag{5.67}$$

Clearly, the second and third equations in Eq. (5.67) are in contradiction with each other. Therefore, ***there exists no second-order time accurate two-field form PCE-GSSSS framework of algorithms when we treat the velocity term explicitly.*** To make sure of this point, we define the local truncation errors for q and ν from $\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n$ as

$$\tau_q^{(2)}(\Delta t) := q(t_{n+1}) - A_{11}q(t_n) - \Delta t A_{12}\nu(t_n) \tag{5.68}$$

$$\tau_\nu^{(2)}(\Delta t) := \nu(t_{n+1}) - \frac{A_{21}}{\Delta t}q(t_n) - A_{22}\nu(t_n) \tag{5.69}$$

Substituting the Taylor series expansion of $q(t_{n+1})$ at time t_n, i.e., Eq. (4.39), into Eq. (5.68) yields

$$\begin{aligned}\tau_q^{(2)}(\Delta t) = {} & \Delta t(1-\lambda_1)\nu(t_n) \\ & + \frac{\Delta t^2}{2}[\dot{\nu}(t_n) + 4\lambda_2\left(1 - W_1\Lambda_4\right)\xi\omega\nu(t_n) \\ & + 2\lambda_2\omega^2 q(t_n)] + \mathcal{O}(\Delta t^3)\end{aligned} \tag{5.70}$$

Therefore, $\tau_q^{(2)}(\Delta t) = \mathcal{O}(\Delta t)$ is obtained for arbitrary algorithmic parameters; $\tau_q^{(2)}(\Delta t) = \mathcal{O}(\Delta t^2)$ is obtained if $\lambda_1 = 1$ and all other algorithmic parameters are arbitrary; and $\tau_q^{(2)}(\Delta t) = \mathcal{O}(\Delta t^3)$ is obtained if

$$\lambda_1 = 1,\ \lambda_2 = \frac{1}{2},\ \text{ and }\ W_1\Lambda_4 = 0 \tag{5.71}$$

and all other algorithmic parameters are arbitrary. Next, substituting the Taylor series expansion of $\nu(t_{n+1})$ at time t_n into Eq. (5.69) yields

$$\begin{aligned}\tau_\nu^{(2)}(\Delta t) = {} & \Delta t\left[\dot{\nu}(t_n) + 2(1 - W_1\Lambda_4)\xi\omega\nu(t_n) + \omega^2 q(t_n)\right] \\ & + \frac{\Delta t^2}{2}\left[\ddot{\nu}(t_n) + 2(W_1\Lambda_1 - W_2\Lambda_2)\omega^2\nu(t_n)\right] + \mathcal{O}(\Delta t^3)\end{aligned} \tag{5.72}$$

Therefore, $\tau_\nu^{(2)}(\Delta t) = \mathcal{O}(\Delta t^2)$ is obtained if $W_1\Lambda_4 = 0$ and all other algorithmic parameters are arbitrary; however, $\tau_\nu^{(2)}(\Delta t) = \mathcal{O}(\Delta t^3)$ cannot be achieved. Hence, only first-order time accurate algorithms exist within the two-field form PCE-GSSSS framework of algorithms with explicit treatment of the velocity term. The

necessary and sufficient conditions for the first-order time accuracy are:

$$\begin{gathered} \lambda_1 = 1,\ \lambda_2 = \frac{1}{2},\ W_1\Lambda_4 = 0 \\ \text{and}\ \ W_2\Lambda_2,\ \Lambda_1 : \textit{arbitrary} \end{gathered} \tag{5.73}$$

It is important to note that, for the undamped case, i.e., $\xi = 0$, the condition, $W_1\Lambda_4 = 0$, is not required; therefore, the second-order time accurate algorithms do exist under the following conditions:

$$\begin{gathered} \lambda_1 = 1,\ \ \lambda_2 = \frac{1}{2} \\ W_1\Lambda_1 - W_2\Lambda_2 = \frac{1}{2} \\ \textbf{(for homogeneous case)} \end{gathered} \tag{5.74}$$

and

$$\begin{gathered} \lambda_1 = 1,\ \ \lambda_2 = \frac{1}{2},\ \ \Lambda_4 = 1,\ \ W_1 = \frac{1}{2} \\ \text{and}\ \ W_2\Lambda_2 = \frac{\Lambda_1 - 1}{2} \\ \textbf{(for non-homogeneous case)} \end{gathered} \tag{5.75}$$

However, it is important to note that once we impose the spectral condition from the two-field form I-GSSSS framework of algorithms, i.e., the classical midpoint rule, we must have $\Lambda_1 = 1$; therefore, the condition given in Eq. (5.75) leads to

$$\begin{gathered} \lambda_1 = \Lambda_1 = 1,\ \ \lambda_2 = \frac{1}{2},\ \ \Lambda_4 = 1,\ \ W_1 = \frac{1}{2} \\ \text{and}\ \ W_2\Lambda_2 = 0 \\ \textbf{(for non-homogeneous case)} \end{gathered} \tag{5.76}$$

and, we see again that a second-order time accurate algorithm does not exist within this framework (see general explicit framework discussed later for exception)).

5.2 E-GSSSS Framework of Algorithms in Two-Field Form

Consider the initial-value problem in two-field form; see Eq. (4.1). To design the form of general explicit GSSSSS algorithms in the two-field form or the ***two-field form E-GSSSS framework of algorithms***, we again consider the following two cases for damped system again: (i) Implicit treatment of the velocity term and (ii) Explicit treatment of the velocity term.

5.2.1 Implicit Treatment of the Velocity Term

Eliminating $\boldsymbol{\nu}_{n+1}$ in $\tilde{\mathbf{q}}$ in Algorithm 1, the integrator for the two-field form E-GSSSS framework of algorithms with the implicit treatment of the velocity term may be given as

$$\mathbf{M}\frac{\boldsymbol{\nu}_{n+1}-\boldsymbol{\nu}_n}{\Delta t}+\mathbf{C}\tilde{\boldsymbol{\nu}}+\mathbf{K}\check{\mathbf{q}}=\tilde{\mathbf{f}} \tag{5.77}$$

where the algorithmic velocity and external force are given as

$$\tilde{\boldsymbol{\nu}} = (1-W_1\Lambda_4)\boldsymbol{\nu}_n+W_1\Lambda_4\boldsymbol{\nu}_{n+1} \tag{5.78}$$

$$\tilde{\mathbf{f}} = (1-W_1)\mathbf{f}_n+W_1\mathbf{f}_{n+1} \ \text{ or } \ \tilde{\mathbf{f}}=\mathbf{f}(t_{n+W_1}) \tag{5.79}$$

respectively. We define $\check{\mathbf{q}}$ as

$$\check{\mathbf{q}}=\mathbf{q}_n+(W_1\Lambda_1-W_2\Lambda_2)\boldsymbol{\nu}_n\Delta t \tag{5.80}$$

The updates are:

$$\begin{aligned}\mathbf{q}_{n+1}&=\mathbf{q}_n+\Delta t\lambda_1\boldsymbol{\nu}_n+\Delta t\lambda_2\Delta\boldsymbol{\nu}\\ \boldsymbol{\nu}_{n+1}&=\boldsymbol{\nu}_n+\Delta\boldsymbol{\nu}\end{aligned} \tag{5.81}$$

The modal decomposition yields:

$$\begin{aligned}\Delta t\nu_{n+1} &= \frac{1}{1+2W_1\Lambda_4\xi\Omega}\,(\,\tilde{g}\Delta t^2-\Omega^2 q_n\\ &\quad+\left[1-2(1-W_1\Lambda_4)\xi\Omega-(W_1\Lambda_1-W_2\Lambda_2)\Omega^2\right]\nu_n\Delta t\,) \qquad (5.82)\\ q_{n+1} &= q_n+\Delta t(\lambda_1-\lambda_2)\nu_n+\Delta t\lambda_2\nu_{n+1} \qquad (5.83)\end{aligned}$$

where $\tilde{g} = (1 - W_1)g_n + W_1 g_{n+1}$ or $\tilde{g} = g(t_{n+W_1})$, and $\Omega := \omega\Delta t$, which leads to the same amplification matrix and the external load vector given in Eq. (5.19) and Eq. (5.20) for $\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n + \mathbf{L}_{n+W_1}$ in which $\mathbf{y}_{n+1} = \{q_{n+1}, \Delta t\nu_{n+1}\}^T$ and $\mathbf{y}_n = \{q_n, \Delta t\nu_n\}^T$.

Order of Time Accuracy

Since the amplification matrix $\mathbf{A} \in \mathbb{R}^{2\times 2}$ and the external load vector $\mathbf{L}_{n+W_1} \in \mathbb{R}^2$ are given exactly in the same form as for the two-field form PCE-GSSSS framework of algorithms with the implicit treatment of the velocity term (see Eq. (5.19) - Eq. (5.20)), the necessary and sufficient conditions for the second-order time accuracy of algorithms for the homogeneous and non-homogeneous cases are given as shown in Eq. (5.31) and Eq. (5.36), i.e.,

$$\boxed{\lambda_1 = 1, \;\; \lambda_2 = \frac{1}{2}, \;\; W_1\Lambda_4 = \frac{1}{2}, \;\; \text{and} \;\; W_1\Lambda_1 - W_2\Lambda_2 = \frac{1}{2}} \tag{5.84}$$

and

$$\boxed{\lambda_1 = 1, \;\; \lambda_2 = \frac{1}{2}, \;\; \Lambda_4 = 1, \;\; W_1 = \frac{1}{2}, \;\; \text{and} \;\; W_2\Lambda_2 = \frac{\Lambda_1 - 1}{2}} \tag{5.85}$$

respectively.

Spectral Analysis

In the E-GSSSS framework of algorithms, it is no longer possible to use $\rho_{i\infty} \in [0, 1]$ for $i = 1, 2$ to control the algorithmic parameters. This is because the principal roots of the amplification matrix tend to become greater than unity in the high-frequency limit, $\Omega_\infty = \lim_{\Delta t\to\infty} \Omega(\Delta t)$, as stated by Dahlquist's theorem [37] which states that there exists no explicit A-stable linear multistep method. Instead of using the principal roots, we employ the spectral radius at the bifurcation limit, $\Omega = \Omega_b$, denoted by ρ_b. Note that the principal roots have the same real value at Ω_b, i.e., $\rho_b = \rho_{1b} = \rho_{2b} \in \mathbb{R}$. At Ω_b, the characteristic equation for $\mathbf{A}$ can be written as

$$(\zeta + \rho_b)^2 = 0 \tag{5.86}$$

Therefore, from a comparison with

$$\zeta^2 - A_1^b \zeta + A_2^b = 0 \tag{5.87}$$

where the principal invariants of $\mathbf{A}(\Omega_b)$, i.e., A_1^b and A_2^b, are given from Eq. (5.21) and Eq. (5.22) with $\Omega = \Omega_b$, we have

$$\begin{aligned} 2\rho_b &= -\frac{2 - 2(1 - 2W_1\Lambda_4)\xi\Omega_b - (W_1\Lambda_1 + \lambda_2 - W_2\Lambda_2)\Omega_b^2}{1 + 2\xi\Omega_b W_1\Lambda_4} \\ \rho_b^2 &= \frac{1 - 2(1 - W_1\Lambda_4)\xi\Omega_b + (\lambda_1 - W_1\Lambda_1 - \lambda_2 + W_2\Lambda_2)\Omega_b^2}{1 + 2\xi\Omega_b W_1\Lambda_4} \end{aligned} \tag{5.88}$$

Imposing the conditions for the second-order time accuracy for non-homogeneous case (Eq. (5.85)) yields

$$\rho_b = \sqrt{\frac{1 - \xi\Omega_b}{1 + \xi\Omega_b}} \tag{5.89}$$

$$\Omega_b = 2\sqrt{1 - \xi^2} \tag{5.90}$$

For the undamped case ($\xi = 0$),

$$\rho_b = 1 \tag{5.91}$$

$$\Omega_b = 2 \tag{5.92}$$

Notice that the stability limit, Ω_s, is given as $\Omega_s = \Omega_b$ if $\rho_b = 1$.

Algorithm 4

Two-field Form E-GSSSS Framework of Algorithms for Linear Dynamical Systems: Implicit Treatment of the velocity term

Integrator:

$$\left[\frac{\mathbf{M}}{\Delta t} + \mathbf{C}W_1\Lambda_4\right]\Delta\boldsymbol{\nu} = \tilde{\mathbf{f}} - \mathbf{K}\check{\mathbf{q}} - \mathbf{C}\boldsymbol{\nu}_n$$

where

$$\begin{aligned} \check{\mathbf{q}} &= \mathbf{q}_n + (W_1\Lambda_1 - W_2\Lambda_2)\boldsymbol{\nu}_n\Delta t \\ \tilde{\mathbf{f}} &= (1 - W_1)\mathbf{f}_n + W_1\mathbf{f}_{n+1} \;\; or \;\; \tilde{\mathbf{f}} = \mathbf{f}(t_{n+W_1}) \end{aligned}$$

Updates:

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \Delta t \lambda_1 \boldsymbol{\nu}_n + \Delta t \lambda_2 \Delta \boldsymbol{\nu}$$
$$\boldsymbol{\nu}_{n+1} = \boldsymbol{\nu}_n + \Delta \boldsymbol{\nu}$$

Initial conditions:

$$\mathbf{q}(t_0) = \mathbf{q}_0$$
$$\boldsymbol{\nu}(t_0) = \boldsymbol{\nu}_0$$

Algorithmic parameters (for second-order time accuracy):

$$W_1 = \frac{1}{2}, \;\; W_2 = \frac{\Lambda_1 - 1}{2\Lambda_2}$$
$$\lambda_1 = 1, \;\; \lambda_2 = \frac{1}{2}, \;\; \Lambda_4 = 1$$

Remark 9 (Algorithm 4)

1. *Algorithm 4 can be also written in the following form:*

$$\mathbf{M}\frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t} + \mathbf{C}\tilde{\boldsymbol{\nu}} + \mathbf{K}\check{\mathbf{q}} = \tilde{\mathbf{f}} \tag{5.93}$$

   where $\tilde{\boldsymbol{\nu}} = (1 - W_1\Lambda_4)\boldsymbol{\nu}_n + W_1\Lambda_4\boldsymbol{\nu}_{n+1}$, *and*

$$\frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t} = (\lambda_1 - \lambda_2)\boldsymbol{\nu}_n + \lambda_2\boldsymbol{\nu}_{n+1} \tag{5.94}$$

2. *The* ***second-order time accurate, conditionally stable*** *member in Algorithm 4 yields:*

$$\begin{aligned}
&\mathbf{M}\frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t} + \mathbf{C}\tilde{\boldsymbol{\nu}} + \mathbf{K}\check{\mathbf{q}} = \tilde{\mathbf{f}} \\
&\check{\mathbf{q}} = \mathbf{q}_n + \frac{\Delta t}{2}\boldsymbol{\nu}_n \\
&\tilde{\boldsymbol{\nu}} = \frac{\boldsymbol{\nu}_{n+1} + \boldsymbol{\nu}_n}{2} = \frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t} \\
&\tilde{\mathbf{f}} = \mathbf{f}(t_{n+1/2})
\end{aligned} \tag{5.95}$$

   for any $\Lambda_i \in \mathbb{R}$ *for* $i = 1, 2$. *Therefore, there exists only one second-order time accurate algorithm in Algorithm 4, and the stability limit or the bifurcation limit is* $\Omega_s = \Omega_b = 2$. *Eq.* (5.95) *is actually equivalent to the original* ***explicit velocity based scheme*** *[38–40].*

3. *The integrator and the updates in the* ***D-form*** *can be written in*

$$\left[\frac{\mathbf{M}}{\lambda_2\Delta t^2}+\mathbf{C}\frac{W_1\Lambda_4}{\lambda_2\Delta t}\right]\Delta\mathbf{q}=\tilde{\mathbf{f}}-\mathbf{K}\check{\mathbf{q}}-\mathbf{C}\left[1-\frac{\lambda_1}{\lambda_2}W_1\Lambda_4\right]\boldsymbol{\nu}_n \tag{5.96}$$

and

$$\begin{aligned}\boldsymbol{\nu}_{n+1}&=\left(1-\frac{\lambda_1}{\lambda_2}\right)\boldsymbol{\nu}_n+\frac{\Delta\mathbf{q}}{\lambda_2\Delta t}\\ \mathbf{q}_{n+1}&=\mathbf{q}_n+\Delta\mathbf{q}\end{aligned} \tag{5.97}$$

respectively.

4. ***Symplecticness:*** *From* $\det(\mathbf{A})=1$ *where* $\mathbf{A}$ *given by Eq.* (5.19) *for the undamped, homogeneous case, the condition for symplectic members is given as*

$$\lambda_1-\lambda_2-W_1\Lambda_1+W_2\Lambda_2=0 \tag{5.98}$$

Hence, the second-order time accurate member given in this framework, i.e., the explicit velocity based scheme [38–40], is symplectic.

5.2.2 Explicit Treatment of the Velocity Term

The integrator for the two-field form E-GSSSS framework of algorithms with the explicit treatment of the velocity term may be given as

$$\mathbf{M}\frac{\boldsymbol{\nu}_{n+1}-\boldsymbol{\nu}_n}{\Delta t}+\mathbf{C}\check{\boldsymbol{\nu}}+\mathbf{K}\check{\mathbf{q}}=\tilde{\mathbf{f}} \tag{5.99}$$

where the external force is given as

$$\tilde{\mathbf{f}}=(1-W_1)\mathbf{f}_n+W_1\mathbf{f}_{n+1}\ \text{ or }\ \tilde{\mathbf{f}}=\mathbf{f}(t_{n+W_1}) \tag{5.100}$$

and $\check{\mathbf{q}}$ and $\check{\boldsymbol{\nu}}$ are defined as

$$\check{\mathbf{q}} = \mathbf{q}_n+(W_1\Lambda_1-W_2\Lambda_2)\boldsymbol{\nu}_n\Delta t \tag{5.101}$$

$$\check{\boldsymbol{\nu}} = (1-W_1\Lambda_4)\boldsymbol{\nu}_n \tag{5.102}$$

respectively. The associated updates are:

$$\begin{aligned} \mathbf{q}_{n+1} &= \mathbf{q}_n + \Delta t \lambda_1 \boldsymbol{\nu}_n + \Delta t \lambda_2 \Delta \boldsymbol{\nu} \\ \boldsymbol{\nu}_{n+1} &= \boldsymbol{\nu}_n + \Delta \boldsymbol{\nu} \end{aligned} \tag{5.103}$$

The modal decomposition yields

$$\begin{aligned} \Delta t \nu_{n+1} &= \tilde{g} \Delta t^2 - \Omega^2 q_n \\ &\quad + \left[1 + 2\xi(W_1\Lambda_4 - 1)\Omega + (W_2\Lambda_2 - W_1\Lambda_1)\Omega^2\right] \nu_n \Delta t \qquad (5.104) \\ q_{n+1} &= q_n + \Delta t(\lambda_1 - \lambda_2)\nu_n + \Delta t \lambda_2 \nu_{n+1} \qquad (5.105) \end{aligned}$$

where $\tilde{g} = (1 - W_1)g_n + W_1 g_{n+1}$ or $\tilde{g} = g(t_{n+W_1})$.

Order of Time Accuracy

The amplification matrix, $\mathbf{A} \in \mathbb{R}^{2\times 2}$, and the external load vector, $\mathbf{L}_{n+W_1} \in \mathbb{R}^2$, take the exact same form as shown in Eq. (5.60) - Eq. (5.61). Recall that there exists no second-order time accurate scheme in this framework for the damped case ($\xi \neq 0$); see Eq. (5.67) in Section 5.1.2. Hence, the conclusion is that it is not possible to design the explicit algorithmic framework of the second-order time accuracy in the two-field form for the damped case with explicit treatment of velocity term unlike the previous case with implicit treatment of the velocity term.

5.3 Unified Algorithmic Framework in the Two-field Form

In this section, we show how we can combine the I-, PCE-, and E-GSSSS frameworks of algorithms in the two-field form within a single unified framework. The main objective of constructing the unified framework is that all schemes within the individual frameworks can be readily derived simply by controlling additional algorithmic parameters. Recall that the implicit midpoint/trapezoidal rule is

the only member which is second-order time accurate and unconditionally stable within Algorithm 1, i.e., the implicit GSSSS framework of algorithms in the sense of the two-field form. We can now show a unified representation without loss of generality following the replacements,

$$\begin{aligned} \Lambda_4 &\rightarrow \Lambda_4 \eta_1 \\ \Lambda_2 &\rightarrow \Lambda_2 \eta_2 \end{aligned} \tag{5.106}$$

where η_1 and η_2 are the additional algorithmic parameters.

Algorithm 5

Unified Representation: Two-Field Form GSSSS Framework of Algorithms for Linear Dynamical Systems

Integrator:

$$\mathbf{M}\frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t} + \mathbf{C}\tilde{\boldsymbol{\nu}} + \mathbf{K}\tilde{\mathbf{q}} = \mathbf{f}(t_{n+W_1})$$

where

$$\begin{aligned} \tilde{\boldsymbol{\nu}} &= \boldsymbol{\nu}_n + W_1 \Lambda_4 \eta_1 \Delta \boldsymbol{\nu} \\ \tilde{\mathbf{q}} &= \mathbf{q}_n + \Delta t W_1 \Lambda_1 \boldsymbol{\nu}_n + \Delta t W_2 \Lambda_2 \eta_2 \Delta \boldsymbol{\nu} \end{aligned}$$

Updates:

$$\begin{aligned} \mathbf{q}_{n+1} &= \mathbf{q}_n + \Delta t \lambda_1 \boldsymbol{\nu}_n + \Delta t \lambda_2 \Delta \boldsymbol{\nu} \\ \boldsymbol{\nu}_{n+1} &= \boldsymbol{\nu}_n + \Delta \boldsymbol{\nu} \end{aligned}$$

Initial conditions:

$$\begin{aligned} \mathbf{q}(t_0) &= \mathbf{q}_0 \\ \boldsymbol{\nu}(t_0) &= \boldsymbol{\nu}_0 \end{aligned}$$

Remark 10 (Algorithm 5)

1. ***Implicit Scheme:*** *When $\eta_1 = \eta_2 = 1$, Algorithm 5 recovers Algorithm 1; note that the **implicit midpoint/trapezoidal rule** is the only existing second-order time accurate unconditionally stable algorithm in this family.*

2. ***Explicit Scheme (with the implicit treatment of the velocity term):*** *When $\eta_1 = 1$ and $\eta_2 = 0$, Algorithm 5 reduces to an explicit algorithm with the implicit treatment of the velocity term. The only member with the second-order time accuracy is the* ***explicit velocity based scheme*** *[38–40] obtained by setting*

$$\lambda_1 = 1, \;\; \lambda_2 = \frac{1}{2}, \;\; W_1\Lambda_4 = \frac{1}{2}, \;\; \textit{and} \;\; W_1\Lambda_1 = \frac{1}{2} \tag{5.107}$$

for the homogeneous case, and

$$\lambda_1 = 1, \;\; \lambda_2 = \frac{1}{2}, \;\; \Lambda_4 = 1, \;\; W_1 = \frac{1}{2}, \;\; \textit{and} \;\; \Lambda_1 = 1 \tag{5.108}$$

for the nonhomogeneous case.

3. ***Explicit Scheme (with the explicit treatment of the velocity term):*** *When $\eta_1 = \eta_2 = 0$, Algorithm 5 reduces to an explicit algorithm with the explicit treatment of the velocity term. However, there exists no second-order time accurate scheme for the damped case in this algorithmic structure; in other words, all schemes are of only first-order time accuracy.*

5.4 PCE-GSSSS Framework of Algorithms in Single-Field Form

Since there exists no useful predictor-corrector schemes in the two-field form, we next turn our focus on the feasible single-field form designs of the predictor-corrector explicit GSSSS algorithms.

5.4.1 Implicit Treatment of the Velocity Term

The basic structure of the predictor-corrector form representation of the single-field form implicit GSSSS framework of algorithms, i.e., the ***single-field form PCE-GSSSS framework of algorithms***, with the implicit treatment of the

velocity term are designed as follows:

Predictor:

$$\hat{\mathbf{q}} = \mathbf{q}_n + \lambda_1 \dot{\mathbf{q}}_n \Delta t + (\lambda_2 - \lambda_3)\ddot{\mathbf{q}}_n \Delta t^2 \tag{5.109}$$

Balance Equation:

$$\mathbf{M}\tilde{\mathbf{a}} + \mathbf{C}\tilde{\mathbf{v}} + \mathbf{K}\breve{\mathbf{q}} = \tilde{\mathbf{f}} \tag{5.110}$$

where

$$\begin{aligned}
\breve{\mathbf{q}} &= \mathbf{q}_n + \frac{W_3\Lambda_3}{\lambda_3}(\hat{\mathbf{q}} - \mathbf{q}_n) \\
&\quad + \left(W_1\Lambda_1 - \frac{\lambda_1}{\lambda_3}W_3\Lambda_3\right)\dot{\mathbf{q}}_n \Delta t + \left(W_2\Lambda_2 - \frac{\lambda_2}{\lambda_3}W_3\Lambda_3\right)\ddot{\mathbf{q}}_n \Delta t^2 && (5.111)\\
\tilde{\mathbf{v}} &= \dot{\mathbf{q}}_n + W_1\Lambda_4\ddot{\mathbf{q}}_n \Delta t + W_2\Lambda_5(\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n)\Delta t && (5.112)\\
\tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1\Lambda_6(\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n) && (5.113)\\
\tilde{\mathbf{f}} &= \mathbf{f}_n + W_1(\mathbf{f}_{n+1} - \mathbf{f}_n) && (5.114)
\end{aligned}$$

Corrector:

$$\begin{aligned}
\mathbf{q}_{n+1} &= \hat{\mathbf{q}} + \lambda_3\ddot{\mathbf{q}}_{n+1}\Delta t^2 && (5.115)\\
\dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5(\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n)\Delta t && (5.116)
\end{aligned}$$

Initial conditions:

$$\begin{aligned}
\mathbf{q}(t_0) &= \mathbf{q}_0 \\
\dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0
\end{aligned} \tag{5.117}$$

Note that the algorithmic external force is given as $\tilde{\mathbf{f}} = \mathbf{f}_{n+W_1}$ or $\tilde{\mathbf{f}} = \mathbf{f}(t_{n+W_1})$. The single-degree-of-freedom system by mode superposition are readily obtained as follows:

Predictor:

$$\hat{q} = q_n + \lambda_1 \dot{q}_n \Delta t + (\lambda_2 - \lambda_3)\ddot{q}_n \Delta t^2 \tag{5.118}$$

Balance Equation:

$$\tilde{a} + 2\xi\omega\tilde{v} + \omega^2\breve{q} = \tilde{g} \tag{5.119}$$

where

$$\begin{aligned}
\breve{q} &= q_n + \frac{W_3\Lambda_3}{\lambda_3}(\hat{q} - q_n) \\
&\quad + \left(W_1\Lambda_1 - \frac{\lambda_1}{\lambda_3}W_3\Lambda_3\right)\dot{q}_n\Delta t + \left(W_2\Lambda_2 - \frac{\lambda_2}{\lambda_3}W_3\Lambda_3\right)\ddot{q}_n\Delta t^2 && (5.120)\\
\tilde{v} &= \dot{q}_n + W_1\Lambda_4\ddot{q}_n\Delta t + W_2\Lambda_5(\ddot{q}_{n+1} - \ddot{q}_n)\Delta t && (5.121)\\
\tilde{a} &= \ddot{q}_n + W_1\Lambda_6(\ddot{q}_{n+1} - \ddot{q}_n) && (5.122)\\
\tilde{g} &= g_n + W_1(g_{n+1} - g_n) \text{ or } g(t_{n+W_1}) && (5.123)
\end{aligned}$$

Corrector:

$$\begin{aligned}
q_{n+1} &= \hat{q} + \lambda_3\ddot{q}_{n+1}\Delta t^2 && (5.124)\\
\dot{q}_{n+1} &= \dot{q}_n + \lambda_4\ddot{q}_n\Delta t + \lambda_5(\ddot{q}_{n+1} - \ddot{q}_n)\Delta t && (5.125)
\end{aligned}$$

Initial conditions:

$$\begin{aligned}
q(t_0) &= q_0 \\
\dot{q}(t_0) &= \dot{q}_0
\end{aligned} \tag{5.126}$$

The algorithmic parameters (W_i, Λ_j, and λ_k for $i = 1, 2, 3$, $j = 1, 2, \cdots, 6$ and $k = 1, 2, \cdots, 5$) are directly employed from Algorithm 2 and Algorithm 3. Eq. (5.118) - Eq. (5.125) can be cast into the following expression:

$$\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n + \mathbf{L}_{n+W_1} \tag{5.127}$$

where the amplification matrix and the load vector can be written as

$$\mathbf{A} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} = \begin{bmatrix} 1 & \lambda_1 & \lambda_2 \\ 0 & 1 & \lambda_4 \\ 0 & 0 & 1 \end{bmatrix} + \begin{pmatrix} \lambda_3 \\ \lambda_5 \\ 1 \end{pmatrix} \begin{pmatrix} \alpha_1 & \alpha_2 & \alpha_3 \end{pmatrix} \tag{5.128}$$

in which

$$\begin{aligned}
\alpha_1 &= -\frac{\Omega^2}{D} \\
\alpha_2 &= -\frac{2\xi\Omega + W_1\Lambda_1\Omega^2}{D} \\
\alpha_3 &= -\frac{1 + 2W_1\Lambda_4\xi\Omega + (W_2\Lambda_2 - W_3\Lambda_3)\,\Omega^2}{D} \\
D &= W_1\Lambda_6 + 2W_2\Lambda_5\xi\Omega \\
\Omega &= \omega\Delta t
\end{aligned} \tag{5.129}$$

and the external load vector is given as

$$\mathbf{L}_{n+W_1} = \frac{\Delta t^2}{D}[(1 - W_1)g_n + W_1 g_{n+1}] \begin{pmatrix} \lambda_3 \\ \lambda_5 \\ 1 \end{pmatrix} \tag{5.130}$$

The characteristic polynomial of the amplification matrix is given as

$$-\det(\zeta\mathbf{I}_3 - \mathbf{A}) = \zeta^3 - A_1\zeta^2 + A_2\zeta - A_3 = 0 \tag{5.131}$$

where ζ are the eigenvalues of $\mathbf{A}$; $\mathbf{I}_3$ denotes the 3×3 identity matrix; and A_i $(i = 1, 2, 3)$ are defined as

$$A_1 = \mathrm{tr}(\mathbf{A}) = \zeta_1 + \zeta_2 + \zeta_3 \tag{5.132}$$

$$A_2 = \frac{1}{2}\left[(\mathrm{tr}(\mathbf{A}))^2 - \mathrm{tr}(\mathbf{A}^2)\right] = \zeta_1\zeta_2 + \zeta_2\zeta_3 + \zeta_3\zeta_1 \tag{5.133}$$

$$A_3 = \det(\mathbf{A}) = \zeta_1\zeta_2\zeta_3 \tag{5.134}$$

Order of Time Accuracy

Again, consider the homogeneous case for simplicity. The difference equation is obtained by

$$\mathbf{y}_{n+2} - A_1\mathbf{y}_{n+1} + A_2\mathbf{y}_n - A_3\mathbf{y}_{n-1} = \mathbf{0} \tag{5.135}$$

We define the local truncation error for q in the sense of the three-stage linear-multistep method as

$$\tau_q^{(1)}(\Delta t) := \frac{1}{\Delta t^2}\left[q(t_{n+2}) - A_1 q(t_{n+1}) + A_2 q(t_n) - A_3 q(t_{n-1})\right] \tag{5.136}$$

where $q(t_{n+2})$, $q(t_{n+1})$, $q(t_n)$, and $q(t_{n-1})$ denote the exact solutions at time t_{n+2}, t_{n+1}, t_n, and t_{n-1}, respectively. Substituting the Taylor series expansions of $q(t_{n+2})$ and $q(t_{n+1})$ at time t_n, i.e., Eq. (4.135) - Eq. (4.137), respectively, into Eq. (5.136) yields

$$\begin{aligned}\tau_q^{(1)}(\Delta t) = \frac{1}{D}\left[\ddot{q}(t_n) + 2\lambda_4\xi\omega\dot{q}(t_n) + \lambda_1\lambda_4\omega^2 q(t_n)\right] \\ + \frac{\Delta t}{D}[W_1\Lambda_6 \dddot{q}(t_n) + 2\left(\lambda_5 + W_1\Lambda_4 - \frac{\lambda_4}{2}\right)\xi\omega\ddot{q}(t_n) \\ + \left(\lambda_2 - \lambda_1\lambda_4 + \lambda_1\lambda_5 + \lambda_4 W_1\Lambda_1\right)\omega^2\dot{q}(t_n)] + \mathcal{O}(\Delta t^2)\end{aligned} \tag{5.137}$$

Hence, for $\tau_q^{(1)}(\Delta t) = \mathcal{O}(\Delta t)$, we must have $\lambda_1 = \lambda_4 = 1$ and all other algorithmic parameters are arbitrary; and for $\tau_q^{(1)}(\Delta t) = \mathcal{O}(\Delta t^2)$, we must have

$$\begin{aligned}&\lambda_1 = \lambda_4 = 1 \\ &W_1\Lambda_6 = \lambda_5 + W_1\Lambda_4 - \frac{\lambda_4}{2} = \lambda_2 - \lambda_1\lambda_4 + \lambda_1\lambda_5 + \lambda_4 W_1\Lambda_1\end{aligned} \tag{5.138}$$

and all other algorithmic parameters are arbitrary. Next, we define the local truncation error for q from $\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n$ as

$$\tau_q^{(2)}(\Delta t) := q(t_{n+1}) - A_{11}q(t_n) - \Delta t A_{12}\dot{q}(t_n) - \Delta t^2 A_{13}\ddot{q}(t_n) \tag{5.139}$$

Substituting the Taylor series expansion of $q(t_{n+1})$ about time t_n, we get

$$\begin{aligned}\tau_q^{(2)}(\Delta t) := \frac{\Delta t}{D}W_1\Lambda_6(1-\lambda_1)\dot{q}(t_n) \\ + \frac{\Delta t^2}{2D}[\left(\lambda_3 + W_1\Lambda_6(\frac{1}{2} - \lambda_2)\right)\ddot{q}(t_n) \\ + 2\left(\lambda_3 - W_2\Lambda_5(\lambda_1 - 1)\right)\xi\omega\dot{q}(t_n) + \lambda_3\omega^2 q(t_n)] + \mathcal{O}(\Delta t^3)\end{aligned} \tag{5.140}$$

Hence, $\tau_q^{(2)}(\Delta t) = \mathcal{O}(\Delta t^2)$ if $\lambda_1 = 1$ and all other algorithmic parameters are arbitrary; and $\tau_q^{(2)}(\Delta t) = \mathcal{O}(\Delta t^3)$ if

$$\lambda_1 = 1 \ \text{ and } \ \lambda_2 = \frac{1}{2} \tag{5.141}$$

and all other algorithmic parameters are arbitrary. Hence, from the requirements for both $\tau_q^{(1)}(\Delta t) = \mathcal{O}(\Delta t^2)$ and $\tau_q^{(2)}(\Delta t) = \mathcal{O}(\Delta t^3)$, the necessary and sufficient

conditions of the second-order time accuracy for the homogeneous case are given as follows:

$$\boxed{\begin{aligned} \lambda_1 &= \lambda_4 = 1 \\ \lambda_2 &= \frac{1}{2} \\ \lambda_5 &= \frac{1}{2}\left[1 + 2W_1(\Lambda_6 - \Lambda_1)\right] \\ \Lambda_1 &= \Lambda_4 \\ \text{All other parameters}: &\quad \text{arbitrary} \\ &\textbf{(for homogeneous case)} \end{aligned}} \tag{5.142}$$

And the necessary and sufficient conditions of the second-order time accuracy for the non-homogeneous case are (detailed analysis is omitted):

$$\boxed{\begin{aligned} \lambda_1 &= \lambda_4 = 1 \\ \lambda_2 &= \frac{1}{2} \\ \lambda_5 &= \frac{1}{2}\left[1 + 2W_1(\Lambda_6 - 1)\right] \\ \Lambda_1 &= \Lambda_4 = 1 \\ \text{All other parameters}: &\quad \text{arbitrary} \\ &\textbf{(for non-homogeneous case)} \end{aligned}} \tag{5.143}$$

Notice that the second-order time accuracy conditions for the single-field form PCE-GSSSS framework of algorithms with the implicit treatment of the velocity term are the same as for the single-field form implicit GSSSS framework of algorithms since the local truncation errors, Eq. (5.137) and Eq. (5.179), have the exactly same forms for both cases. It should be noted that λ_3 does not influence the order of accuracy of the algorithms. Hence, let us introduce a scalar parameter η_3 for the predictor and the corrector by the following replacement

$$\lambda_3 \rightarrow \lambda_3 \eta_3 \tag{5.144}$$

where we let η_3 take on "1" or "0."

Algorithm 6

Single-field Form PCE-GSSSS U0-Family of Algorithms for Linear Dynamical Systems: Implicit Treatment of the Velocity Term

Predictor:

$$\hat{\mathbf{q}} = \mathbf{q}_n + \lambda_1 \dot{\mathbf{q}}_n \Delta t + (\lambda_2 - \lambda_3 \eta_3) \ddot{\mathbf{q}}_n \Delta t^2$$

Balance Equation:

$$\mathbf{M}\tilde{\mathbf{a}} + \mathbf{C}\tilde{\mathbf{v}} + \mathbf{K}\breve{\mathbf{q}} = \tilde{\mathbf{f}}$$

where

$$\begin{aligned}
\breve{\mathbf{q}} &= \mathbf{q}_n + \frac{W_3 \Lambda_3}{\lambda_3}(\hat{\mathbf{q}} - \mathbf{q}_n) \\
&\quad + \left(W_1 \Lambda_1 - \frac{\lambda_1}{\lambda_3} W_3 \Lambda_3 \right) \dot{\mathbf{q}}_n \Delta t + \left(W_2 \Lambda_2 - \frac{\lambda_2}{\lambda_3} W_3 \Lambda_3 \right) \ddot{\mathbf{q}}_n \Delta t^2 \\
\tilde{\mathbf{v}} &= \dot{\mathbf{q}}_n + W_1 \Lambda_4 \ddot{\mathbf{q}}_n \Delta t + W_2 \Lambda_5 (\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n) \Delta t \\
\tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1 \Lambda_6 (\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n) \\
\tilde{\mathbf{f}} &= \mathbf{f}_n + W_1 (\mathbf{f}_{n+1} - \mathbf{f}_n)
\end{aligned}$$

Corrector:

$$\begin{aligned}
\mathbf{q}_{n+1} &= \hat{\mathbf{q}} + \lambda_3 \eta_3 \ddot{\mathbf{q}}_{n+1} \Delta t^2 \\
\dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4 \ddot{\mathbf{q}}_n \Delta t + \lambda_5 (\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n) \Delta t
\end{aligned}$$

Initial conditions:

$$\begin{aligned}
\mathbf{q}(t_0) &= \mathbf{q}_0 \\
\dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0
\end{aligned}$$

Algorithmic parameters:

$$
\begin{aligned}
W_1\Lambda_1 &= \frac{1}{1+\rho_\infty^{\mathrm{s}}} \quad , \quad \lambda_1 = 1 \\
W_2\Lambda_2 &= \frac{1}{2(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_2 = \frac{1}{2} \\
W_3\Lambda_3 &= \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_3 = \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \\
W_1\Lambda_4 &= \frac{1}{1+\rho_\infty^{\mathrm{s}}} \quad , \quad \lambda_4 = 1 \\
W_2\Lambda_5 &= \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_5 = \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \\
W_1\Lambda_6 &= \frac{2+\rho_\infty^{\min}+\rho_\infty^{\max}+\rho_\infty^{\mathrm{s}}-\rho_\infty^{\min}\rho_\infty^{\max}\rho_\infty^{\mathrm{s}}}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})}
\end{aligned}
$$

Algorithm 7

Single-field Form PCE-GSSSS V0-Family of Algorithms for Linear Dynamical Systems: Implicit Treatment of the Velocity Term

Predictor:

$$\hat{\mathbf{q}} = \mathbf{q}_n + \lambda_1 \dot{\mathbf{q}}_n \Delta t + (\lambda_2 - \lambda_3\eta_3)\ddot{\mathbf{q}}_n \Delta t^2$$

Balance Equation:

$$\mathbf{M}\tilde{\mathbf{a}} + \mathbf{C}\tilde{\mathbf{v}} + \mathbf{K}\breve{\mathbf{q}} = \tilde{\mathbf{f}}$$

where

$$
\begin{aligned}
\breve{\mathbf{q}} &= \mathbf{q}_n + W_3(\hat{\mathbf{q}} - \mathbf{q}_n) + (W_1 - W_3)\,\dot{\mathbf{q}}_n \Delta t \\
\tilde{\mathbf{v}} &= \dot{\mathbf{q}}_n + W_1\Lambda_4 \ddot{\mathbf{q}}_n \Delta t + W_2\Lambda_5(\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n)\Delta t \\
\tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1\Lambda_6(\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n) \\
\tilde{\mathbf{f}} &= \mathbf{f}_n + W_1(\mathbf{f}_{n+1} - \mathbf{f}_n)
\end{aligned}
$$

Corrector:

$$
\begin{aligned}
\mathbf{q}_{n+1} &= \hat{\mathbf{q}} + \lambda_3\eta_3 \ddot{\mathbf{q}}_{n+1} \Delta t^2 \\
\dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4 \ddot{\mathbf{q}}_n \Delta t + \lambda_5(\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n)\Delta t
\end{aligned}
$$

Initial conditions:

$$\mathbf{q}(t_0) = \mathbf{q}_0$$
$$\dot{\mathbf{q}}(t_0) = \dot{\mathbf{q}}_0$$

Algorithmic parameters:

$$W_1\Lambda_1 = \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \quad , \quad \lambda_1 = 1$$
$$W_2\Lambda_2 = \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \quad , \quad \lambda_2 = \frac{1}{2}$$
$$W_3\Lambda_3 = \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_3 = \frac{1}{2(1+\rho_\infty^{\mathrm{s}})}$$
$$W_1\Lambda_4 = \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \quad , \quad \lambda_4 = 1$$
$$W_2\Lambda_5 = \frac{2}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_5 = \frac{1}{1+\rho_\infty^{\mathrm{s}}}$$
$$W_1\Lambda_6 = \frac{2+\rho_\infty^{\min}+\rho_\infty^{\max}+\rho_\infty^{\mathrm{s}}-\rho_\infty^{\min}\rho_\infty^{\max}\rho_\infty^{\mathrm{s}}}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})}$$

Remark 11 (Algorithms 6 and 7)

1. *Algorithms 6 and 7 are the predictor-corrector explicit (PCE) representations of the implicit GSSSS U0-family and V0-family of algorithms, i.e., Algorithms 2 and 3, respectively, with the implicit treatment of the velocity term. The algorithmic parameters for Algorithms 6 and 7 are directly employed from Algorithms 2 and 3, respectively.*

2. *All members in Algorithms 6 and 7 are second-order time accurate. No member in Algorithms 6 and 7 can conserve the mechanical energy of the system exactly.*

3. *When selecting $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\mathrm{s}} = 0$ with $\eta_3 = 0$ in Algorithm 6, we can recover the* ***central difference method*** *(explicit Newmark method with $\gamma = 1/2$), which is equivalent to the so-called* ***velocity verlet algorithm*** *[41].*

5.4.2 Explicit Treatment of the Velocity Term

In contrast, the predictor-corrector representation of Algorithm 2 and Algorithm 2 with the explicit treatment of the velocity term are designed as follows:

Predictor:

$$\hat{\mathbf{q}} = \mathbf{q}_n + \lambda_1 \dot{\mathbf{q}}_n \Delta t + (\lambda_2 - \lambda_3)\ddot{\mathbf{q}}_n \Delta t^2 \tag{5.145}$$

$$\hat{\mathbf{v}} = \dot{\mathbf{q}}_n + (\lambda_4 - \lambda_5)\ddot{\mathbf{q}}_n \Delta t \tag{5.146}$$

Balance Equation:

$$\mathbf{M}\tilde{\mathbf{a}} + \mathbf{C}\breve{\mathbf{v}} + \mathbf{K}\breve{\mathbf{q}} = \tilde{\mathbf{f}} \tag{5.147}$$

where

$$\begin{aligned}\breve{\mathbf{q}} &= \mathbf{q}_n + \frac{W_3\Lambda_3}{\lambda_3}(\hat{\mathbf{q}} - \mathbf{q}_n) \\ &\quad + \left(W_1\Lambda_1 - \frac{\lambda_1}{\lambda_3}W_3\Lambda_3\right)\dot{\mathbf{q}}_n \Delta t + \left(W_2\Lambda_2 - \frac{\lambda_2}{\lambda_3}W_3\Lambda_3\right)\ddot{\mathbf{q}}_n \Delta t^2 \quad (5.148)\\ \breve{\mathbf{v}} &= \dot{\mathbf{q}}_n + \frac{W_2\Lambda_5}{\lambda_5}(\hat{\mathbf{v}} - \dot{\mathbf{q}}_n) + \left(W_1\Lambda_4 - \frac{\lambda_4}{\lambda_5}W_2\Lambda_5\right)\ddot{\mathbf{q}}_n \Delta t \quad (5.149)\\ \tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1\Lambda_6(\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n) \quad (5.150)\\ \tilde{\mathbf{f}} &= \mathbf{f}_n + W_1(\mathbf{f}_{n+1} - \mathbf{f}_n) \text{ or } \mathbf{f}(t_{n+W_1}) \quad (5.151)\end{aligned}$$

Corrector:

$$\mathbf{q}_{n+1} = \hat{\mathbf{q}} + \lambda_3 \ddot{\mathbf{q}}_{n+1} \Delta t^2 \tag{5.152}$$

$$\dot{\mathbf{q}}_{n+1} = \hat{\mathbf{v}} + \lambda_5 \ddot{\mathbf{q}}_{n+1} \Delta t \tag{5.153}$$

Initial conditions:

$$\begin{aligned}\mathbf{q}(t_0) &= \mathbf{q}_0 \\ \dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0\end{aligned} \tag{5.154}$$

Note that the discrete acceleration $\ddot{\mathbf{q}}_{n+1}$ is given by

$$\ddot{\mathbf{q}}_{n+1} = \frac{1}{W_1\Lambda_6}\left[\mathbf{M}^{-1}\left(\tilde{\mathbf{f}} - \mathbf{C}\breve{\mathbf{v}} - \mathbf{K}\breve{\mathbf{q}}\right) - (1 - W_1\Lambda_6)\ddot{\mathbf{q}}_n\right] \tag{5.155}$$

The single-degree-of-freedom system representation by mode superposition is readily obtained as follows:

Predictor:

$$\hat{q} = q_n + \lambda_1 \dot{q}_n \Delta t + (\lambda_2 - \lambda_3)\ddot{q}_n \Delta t^2 \tag{5.156}$$

$$\hat{v} = \dot{q}_n + (\lambda_4 - \lambda_5)\ddot{q}_n \Delta t \tag{5.157}$$

Balance Equation:

$$\tilde{a} + 2\xi\omega\breve{v} + \omega^2\breve{q} = \tilde{g} \tag{5.158}$$

where

$$\begin{aligned}\breve{q} &= q_n + \frac{W_3\Lambda_3}{\lambda_3}(\hat{q} - q_n) \\ &\quad + \left(W_1\Lambda_1 - \frac{\lambda_1}{\lambda_3}W_3\Lambda_3\right)\dot{q}_n\Delta t + \left(W_2\Lambda_2 - \frac{\lambda_2}{\lambda_3}W_3\Lambda_3\right)\ddot{q}_n\Delta t^2 \qquad (5.159)\\ \breve{v} &= \dot{q}_n + \frac{W_2\Lambda_5}{\lambda_5}(\hat{v} - \dot{q}_n) + \left(W_1\Lambda_4 - \frac{\lambda_4}{\lambda_5}W_2\Lambda_5\right)\ddot{q}_n\Delta t \qquad (5.160)\\ \tilde{a} &= \ddot{\mathbf{q}}_n + W_1\Lambda_6(\ddot{q}_{n+1} - \ddot{q}_n) \qquad (5.161)\\ \tilde{g} &= \mathbf{f}_n + W_1(g_{n+1} - g_n) \text{ or } g(t_{n+W_1}) \qquad (5.162)\end{aligned}$$

Corrector:

$$q_{n+1} = \hat{q} + \lambda_3\ddot{q}_{n+1}\Delta t^2 \tag{5.163}$$

$$\dot{q}_{n+1} = \hat{v} + \lambda_5\ddot{q}_{n+1}\Delta t \tag{5.164}$$

Initial conditions:

$$\begin{aligned}q(t_0) &= q_0 \\ \dot{q}(t_0) &= \dot{q}_0\end{aligned} \tag{5.165}$$

The algorithmic parameters (W_i, Λ_j, and λ_k for $i = 1, 2, 3$, $j = 1, 2, \cdots, 6$ and $k = 1, 2, \cdots, 5$) are directly employed from Algorithm 2 and Algorithm 3. Eq. (5.156) - Eq. (5.164) can be cast into the following expression:

$$\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n + \mathbf{L}_{n+W_1} \tag{5.166}$$

where the amplification matrix and the load vector can be written as

$$\mathbf{A} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} = \begin{bmatrix} 1 & \lambda_1 & \lambda_2 \\ 0 & 1 & \lambda_4 \\ 0 & 0 & 1 \end{bmatrix} + \begin{pmatrix} \lambda_3 \\ \lambda_5 \\ 1 \end{pmatrix} \begin{pmatrix} \alpha_1 & \alpha_2 & \alpha_3 \end{pmatrix} \tag{5.167}$$

in which

$$\begin{aligned} \alpha_1 &= -\frac{\Omega^2}{D} \\ \alpha_2 &= -\frac{2\xi\Omega + W_1\Lambda_1\Omega^2}{D} \\ \alpha_3 &= -\frac{1 + 2\left(W_1\Lambda_4 - W_2\Lambda_5\right)\xi\Omega + \left(W_2\Lambda_2 - W_3\Lambda_3\right)\Omega^2}{D} \\ D &= W_1\Lambda_6 \\ \Omega &= \omega\Delta t \end{aligned} \tag{5.168}$$

and the external load vector is given as

$$\mathbf{L}_{n+W_1} = \frac{\Delta t^2}{D}[(1 - W_1)g_n + W_1 g_{n+1}] \begin{pmatrix} \lambda_3 \\ \lambda_5 \\ 1 \end{pmatrix} \tag{5.169}$$

The characteristic polynomial of the amplification matrix is given as

$$-\det(\zeta\mathbf{I}_3 - \mathbf{A}) = \zeta^3 - A_1\zeta^2 + A_2\zeta - A_3 = 0 \tag{5.170}$$

where ζ are the eigenvalues of $\mathbf{A}$; $\mathbf{I}_3$ denotes the 3×3 identity matrix; and A_i $(i = 1, 2, 3)$ are defined as

$$A_1 = \operatorname{tr}(\mathbf{A}) = \zeta_1 + \zeta_2 + \zeta_3 \tag{5.171}$$

$$A_2 = \frac{1}{2}\left[(\operatorname{tr}(\mathbf{A}))^2 - \operatorname{tr}(\mathbf{A}^2)\right] = \zeta_1\zeta_2 + \zeta_2\zeta_3 + \zeta_3\zeta_1 \tag{5.172}$$

$$A_3 = \det(\mathbf{A}) = \zeta_1\zeta_2\zeta_3 \tag{5.173}$$

Order of Time Accuracy

Again, consider the homogeneous case for simplicity. The difference equation is obtained as

$$\mathbf{y}_{n+2} - A_1\mathbf{y}_{n+1} + A_2\mathbf{y}_n - A_3\mathbf{y}_{n-1} = \mathbf{0} \tag{5.174}$$

We define the local truncation error for q in the sense of the three-stage linear-multistep method as

$$\tau_q^{(1)}(\Delta t) := \frac{1}{\Delta t^2}\left[q(t_{n+2}) - A_1 q(t_{n+1}) + A_2 q(t_n) - A_3 q(t_{n-1})\right] \tag{5.175}$$

where $q(t_{n+2})$, $q(t_{n+1})$, $q(t_n)$, and $q(t_{n-1})$ denote the exact solutions at time t_{n+2}, t_{n+1}, t_n, and t_{n-1}, respectively. Substituting the Taylor series expansions of $q(t_{n+2})$ and $q(t_{n+1})$ at time t_n, i.e., Eq. (4.135) - Eq. (4.137), respectively, into Eq. (5.175) yields

$$\begin{aligned}\tau_q^{(1)}(\Delta t) = {} & \frac{1}{D}\left[\ddot{q}(t_n) + 2\lambda_4\xi\omega\dot{q}(t_n) + \lambda_1\lambda_4\omega^2 q(t_n)\right] \\ & + \frac{\Delta t}{D}[W_1\Lambda_6 \dddot{q}(t_n) + 2\left(\lambda_5 + W_1\Lambda_4 - \frac{\lambda_4}{2} - W_2\Lambda_5\right)\xi\omega\ddot{q}(t_n) \\ & + \left(\lambda_2 - \lambda_1\lambda_4 + \lambda_1\lambda_5 + \lambda_4 W_1\Lambda_1\right)\omega^2\dot{q}(t_n)] + \mathcal{O}(\Delta t^2)\end{aligned} \tag{5.176}$$

Hence, for $\tau_q^{(1)}(\Delta t) = \mathcal{O}(\Delta t)$, we must have $\lambda_1 = \lambda_4 = 1$ and all other algorithmic parameters are arbitrary; and for $\tau_q^{(1)}(\Delta t) = \mathcal{O}(\Delta t^2)$, we must have

$$\begin{aligned}&\lambda_1 = \lambda_4 = 1 \\ &W_1\Lambda_6 = \lambda_5 + W_1\Lambda_4 - \frac{\lambda_4}{2} - W_2\Lambda_5 = \lambda_2 - \lambda_1\lambda_4 + \lambda_1\lambda_5 + \lambda_4 W_1\Lambda_1\end{aligned} \tag{5.177}$$

and all other algorithmic parameters are arbitrary. Next, we define the local truncation error for q from $\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n$ as

$$\tau_q^{(2)}(\Delta t) := q(t_{n+1}) - A_{11} q(t_n) - \Delta t A_{12}\dot{q}(t_n) - \Delta t^2 A_{13}\ddot{q}(t_n) \tag{5.178}$$

Substituting the Taylor series expansion of $q(t_{n+1})$ about time t_n, we get

$$\begin{aligned}\tau_q^{(2)}(\Delta t) := {} & \frac{\Delta t}{D} W_1\Lambda_6(1-\lambda_1)\dot{q}(t_n) \\ & + \frac{\Delta t^2}{2D}[\left(\lambda_3 + W_1\Lambda_6(\frac{1}{2} - \lambda_2)\right)\ddot{q}(t_n) \\ & + 2\lambda_3\xi\omega\dot{q}(t_n) + \lambda_3\omega^2 q(t_n)] + \mathcal{O}(\Delta t^3)\end{aligned} \tag{5.179}$$

Hence, $\tau_q^{(2)}(\Delta t) = \mathcal{O}(\Delta t^2)$ if $\lambda_1 = 1$ and all other algorithmic parameters are arbitrary; and $\tau_q^{(2)}(\Delta t) = \mathcal{O}(\Delta t^3)$ if

$$\lambda_1 = 1 \;\text{ and }\; \lambda_2 = \frac{1}{2} \tag{5.180}$$

and all other algorithmic parameters are arbitrary. Hence, from the requirements for both $\tau_q^{(1)}(\Delta t) = \mathcal{O}(\Delta t^2)$ and $\tau_q^{(2)}(\Delta t) = \mathcal{O}(\Delta t^3)$, the necessary and sufficient conditions of second-order time accuracy for the homogeneous case are given as follows:

$$\boxed{\begin{aligned} \lambda_1 &= \lambda_4 = 1 \\ \lambda_2 &= \frac{1}{2} \\ \lambda_5 &= \frac{1}{2}\left[1 + 2W_1(\Lambda_6 - \Lambda_1)\right] \\ W_1(\Lambda_4 - \Lambda_1) &= W_2\Lambda_5 \\ \text{All other parameters}: &\quad \text{arbitrary} \\ &\textbf{(for homogeneous case)} \end{aligned}} \tag{5.181}$$

And, the necessary and sufficient conditions of second-order time accuracy for the non-homogeneous case are (detailed analysis is omitted):

$$\boxed{\begin{aligned} \lambda_1 &= \lambda_4 = 1 \\ \lambda_2 &= \frac{1}{2} \\ \lambda_5 &= \frac{1}{2}\left[1 + 2W_1(\Lambda_6 - 1)\right] \\ W_1(\Lambda_4 - 1) &= W_2\Lambda_5 \\ \Lambda_1 &= 1 \\ \text{All other parameters}: &\quad \text{arbitrary} \\ &\textbf{(for non-homogeneous case)} \end{aligned}} \tag{5.182}$$

Notice that the second-order time accuracy conditions for the single-field form PCE-GSSSS framework of algorithms with the explicit treatment of the velocity term are different from the conditions for the single-field form implicit GSSSS framework of algorithms, Algorithms 2 and 3. That is, we MUST ensure

$$\boxed{\lambda_5 \neq \Lambda_5 = 0} \tag{5.183}$$

to maintain the second-order time accuracy due to $\Lambda_4 = 1$ from the single-field form Algorithms 2 and 3; however, $\Lambda_5 = 0$ is not acceptable for Algorithms 2 and

3 since it causes conditional stability issues. Hence, we conclude that there exists no second-order time accurate scheme if we treat the velocity term explicitly in the single-field form unless otherwise we impose the condition given in Eq. (5.183), i.e., $\lambda_5 \neq \Lambda_5 = 0$.

If we impose this condition, all schemes in the family of the PCE-GSSSS algorithms with the explicit treatment of the velocity term also become second-order time accurate. Similar to the case of the implicit treatment of the velocity term, λ_3 does not influence the order of accuracy of the algorithms. Hence, we can introduce the scalar parameter η_3 for the predictor and the corrector by the following replacement

$$\lambda_3 \rightarrow \lambda_3 \eta_3 \tag{5.184}$$

Algorithm 8

Single-field Form PCE-GSSSS U0-Family of Algorithms for Linear Dynamical Systems: Explicit Treatment of the Velocity Term

Predictor:

$$\begin{aligned}
\hat{\mathbf{q}} &= \mathbf{q}_n + \lambda_1 \dot{\mathbf{q}}_n \Delta t + (\lambda_2 - \lambda_3 \eta_3) \ddot{\mathbf{q}}_n \Delta t^2 \\
\hat{\mathbf{v}} &= \dot{\mathbf{q}}_n + (\lambda_4 - \lambda_5) \ddot{\mathbf{q}}_n \Delta t
\end{aligned}$$

Balance Equation:

$$\mathbf{M}\tilde{\mathbf{a}} + \mathbf{C}\breve{\mathbf{v}} + \mathbf{K}\breve{\mathbf{q}} = \tilde{\mathbf{f}}$$

where

$$\begin{aligned}
\breve{\mathbf{q}} &= \mathbf{q}_n + \frac{W_3 \Lambda_3}{\lambda_3}(\hat{\mathbf{q}} - \mathbf{q}_n) \\
&\quad + \left(W_1 \Lambda_1 - \frac{\lambda_1}{\lambda_3} W_3 \Lambda_3 \right) \dot{\mathbf{q}}_n \Delta t + \left(W_2 \Lambda_2 - \frac{\lambda_2}{\lambda_3} W_3 \Lambda_3 \right) \ddot{\mathbf{q}}_n \Delta t^2 \\
\breve{\mathbf{v}} &= \dot{\mathbf{q}}_n + \frac{W_2 \Lambda_5}{\lambda_5}(\hat{\mathbf{v}} - \dot{\mathbf{q}}_n) + \left(W_1 \Lambda_4 - \frac{\lambda_4}{\lambda_5} W_2 \Lambda_5 \right) \ddot{\mathbf{q}}_n \Delta t \\
\tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1 \Lambda_6 (\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n) \\
\tilde{\mathbf{f}} &= \mathbf{f}_n + W_1 (\mathbf{f}_{n+1} - \mathbf{f}_n) \;\; or \;\; \mathbf{f}(t_{n+W_1})
\end{aligned}$$

Corrector:

$$\begin{aligned}\mathbf{q}_{n+1} &= \hat{\mathbf{q}} + \lambda_3\eta_3\ddot{\mathbf{q}}_{n+1}\Delta t^2 \\ \dot{\mathbf{q}}_{n+1} &= \hat{\mathbf{v}} + \lambda_5\ddot{\mathbf{q}}_{n+1}\Delta t\end{aligned}$$

Initial conditions:

$$\begin{aligned}\mathbf{q}(t_0) &= \mathbf{q}_0 \\ \dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0\end{aligned}$$

Algorithmic parameters:

$$\begin{aligned} W_1\Lambda_1 &= \frac{1}{1+\rho_\infty^{\mathrm{s}}} \quad , \quad \lambda_1 = 1 \\ W_2\Lambda_2 &= \frac{1}{2(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_2 = \frac{1}{2} \\ W_3\Lambda_3 &= \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_3 = \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \\ W_1\Lambda_4 &= \frac{1}{1+\rho_\infty^{\mathrm{s}}} \quad , \quad \lambda_4 = 1 \\ \Lambda_5 &= 0 \quad , \quad \lambda_5 = \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \end{aligned}$$

$$W_1\Lambda_6 = \frac{2+\rho_\infty^{\min}+\rho_\infty^{\max}+\rho_\infty^{\mathrm{s}}-\rho_\infty^{\min}\rho_\infty^{\max}\rho_\infty^{\mathrm{s}}}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})}$$

Algorithm 9

Single-field Form PCE-GSSSS V0-Family of Algorithms for Linear Dynamical Systems: Explicit Treatment of the Velocity Term

Predictor:

$$\begin{aligned}\hat{\mathbf{q}} &= \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + (\lambda_2 - \lambda_3\eta_3)\ddot{\mathbf{q}}_n\Delta t^2 \\ \hat{\mathbf{v}} &= \dot{\mathbf{q}}_n + (\lambda_4 - \lambda_5)\ddot{\mathbf{q}}_n\Delta t\end{aligned}$$

Balance Equation:

$$\mathbf{M}\tilde{\mathbf{a}} + \mathbf{C}\breve{\mathbf{v}} + \mathbf{K}\breve{\mathbf{q}} = \tilde{\mathbf{f}}$$

where

$$
\begin{aligned}
\breve{\mathbf{q}} &= \mathbf{q}_n + \frac{W_3\Lambda_3}{\lambda_3}(\hat{\mathbf{q}} - \mathbf{q}_n) \\
&\quad + \left(W_1\Lambda_1 - \frac{\lambda_1}{\lambda_3}W_3\Lambda_3\right)\dot{\mathbf{q}}_n\Delta t + \left(W_2\Lambda_2 - \frac{\lambda_2}{\lambda_3}W_3\Lambda_3\right)\ddot{\mathbf{q}}_n\Delta t^2 \\
\breve{\mathbf{v}} &= \dot{\mathbf{q}}_n + \frac{W_2\Lambda_5}{\lambda_5}(\hat{\mathbf{v}} - \dot{\mathbf{q}}_n) + \left(W_1\Lambda_4 - \frac{\lambda_4}{\lambda_5}W_2\Lambda_5\right)\ddot{\mathbf{q}}_n\Delta t \\
\tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1\Lambda_6(\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n) \\
\tilde{\mathbf{f}} &= \mathbf{f}_n + W_1(\mathbf{f}_{n+1} - \mathbf{f}_n) \;\; or \;\; \mathbf{f}(t_{n+W_1})
\end{aligned}
$$

Corrector:

$$
\begin{aligned}
\mathbf{q}_{n+1} &= \hat{\mathbf{q}} + \lambda_3\eta_3\ddot{\mathbf{q}}_{n+1}\Delta t^2 \\
\dot{\mathbf{q}}_{n+1} &= \hat{\mathbf{v}} + \lambda_5\ddot{\mathbf{q}}_{n+1}\Delta t
\end{aligned}
$$

Initial conditions:

$$
\begin{aligned}
\mathbf{q}(t_0) &= \mathbf{q}_0 \\
\dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0
\end{aligned}
$$

Algorithmic parameters:

$$
\begin{aligned}
W_1\Lambda_1 &= \frac{3 + \rho_\infty^{\min} + \rho_\infty^{\max} - \rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \quad , \quad \lambda_1 = 1 \\
W_2\Lambda_2 &= \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \quad , \quad \lambda_2 = \frac{1}{2} \\
W_3\Lambda_3 &= \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_3 = \frac{1}{2(1+\rho_\infty^{\mathrm{s}})} \\
W_1\Lambda_4 &= \frac{3 + \rho_\infty^{\min} + \rho_\infty^{\max} - \rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \quad , \quad \lambda_4 = 1 \\
\Lambda_5 &= 0 \quad , \quad \lambda_5 = \frac{1}{1+\rho_\infty^{\mathrm{s}}} \\
W_1\Lambda_6 &= \frac{2 + \rho_\infty^{\min} + \rho_\infty^{\max} + \rho_\infty^{\mathrm{s}} - \rho_\infty^{\min}\rho_\infty^{\max}\rho_\infty^{\mathrm{s}}}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})}
\end{aligned}
$$

Remark 12 (Algorithms 8 and 9)

1. *Algorithms 8 and 9 are the predictor-corrector explicit (PCE) representations of the explicit GSSSS U0-family and V0-family of algorithms, i.e.,*

Algorithms 2 and 3, respectively, with the explicit treatment of the velocity term. The algorithmic parameters for Algorithms 8 and 9 are directly employed from Algorithms 2 and 3, respectively.

2. *All members in Algorithms 8 and 9 are second-order time accurate. No member in Algorithms 8 and 9 can conserve the mechanical energy of the system exactly.*

3. *When selecting* $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ *and* $\rho_\infty^{\mathrm{s}} = 0$ *with* $\eta_3 = 0$ *in Algorithm 8, we can recover the explicit treatment version of the* ***central difference method*** *(or equivalently, the explicit treatment version of the velocity verlet algorithm [41]).*

5.5 E-GSSSS Framework of Algorithms in Single-Field Form

Consider the initial-value problem in the single-field form; see Eq. (4.1). To design the general form of the explicit GSSSS algorithm structure in the single-field form or the ***single-field form E-GSSSS framework of algorithms***, we again consider the two cases of the implicit and explicit treatment of the velocity term.

5.5.1 Implicit Treatment of the Velocity Term

From the framework of the single-field form I-GSSSS framework of algorithms, we propose the balance equation of the single-field form E-GSSSS framework of algorithms with the implicit treatment of the velocity term as follows:

$$\mathbf{M}\tilde{\mathbf{a}} + \mathbf{C}\tilde{\mathbf{v}} + \mathbf{K}\check{\mathbf{q}} = \tilde{\mathbf{f}} \tag{5.185}$$

where

$$\check{\mathbf{q}} = \mathbf{q}_n + W_1\Lambda_1\dot{\mathbf{q}}_n\Delta t + (W_2\Lambda_2 - W_3\Lambda_3)\ddot{\mathbf{q}}_n\Delta t^2 \tag{5.186}$$

$$\tilde{\mathbf{v}} = \dot{\mathbf{q}}_n + W_1\Lambda_4\ddot{\mathbf{q}}_n\Delta t + W_2\Lambda_5\Delta\mathbf{a}\Delta t \tag{5.187}$$

$$\tilde{\mathbf{a}} = \ddot{\mathbf{q}}_n + W_1\Lambda_6\Delta\mathbf{a} \tag{5.188}$$

$$\tilde{\mathbf{f}} = \mathbf{f}_n + W_1(\mathbf{f}_{n+1} - \mathbf{f}_n) \text{ or } \mathbf{f}(t_{n+W_1}) \tag{5.189}$$

And the associated updates are:

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + \lambda_3\Delta\mathbf{a}\Delta t^2 \tag{5.190}$$

$$\dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5\Delta\mathbf{a}\Delta t \tag{5.191}$$

$$\ddot{\mathbf{q}}_{n+1} = \ddot{\mathbf{q}}_n + \Delta\mathbf{a} \tag{5.192}$$

The above expression is in the A-form (likewise the V- and D-forms may be constructed); that is, we find $\Delta\mathbf{a}$ as the primary unknown from the balance equation

$$[W_1\Lambda_6\mathbf{M} + W_2\Lambda_5\mathbf{C}\Delta t]\,\Delta\mathbf{a} = \tilde{\mathbf{f}} - \mathbf{M}\ddot{\mathbf{q}}_n - \mathbf{C}\,[\dot{\mathbf{q}}_n + W_1\Lambda_4\ddot{\mathbf{q}}_n\Delta t] - \mathbf{K}\check{\mathbf{q}} \tag{5.193}$$

and then compute $\mathbf{q}_{n+1}$, $\dot{\mathbf{q}}_{n+1}$, and $\ddot{\mathbf{q}}_{n+1}$ for $n \in \{0, 1, 2, \cdots, N\}$ with given initial conditions, $\mathbf{q}(t_0) = \mathbf{q}_0$ and $\dot{\mathbf{q}}(t_0) = \dot{\mathbf{q}}_0$. The modal decomposition for the new general explicit framework yields

Integrator:

$$\begin{aligned}&[W_1\Lambda_6 + 2W_2\Lambda_5\xi\omega\Delta t]\,\Delta a\\ &= \tilde{g} - \ddot{q}_n - 2\xi\omega\,[\dot{q}_n + W_1\Lambda_4\ddot{q}_n\Delta t]\\ &- \omega^2\left[q_n + W_1\Lambda_1\dot{q}_n\Delta t + (W_2\Lambda_2 - W_3\Lambda_3)\ddot{q}_n\Delta t^2\right]\end{aligned}$$

Updates:

$$\begin{aligned}q_{n+1} &= q_n + \lambda_1\dot{q}_n\Delta t + \lambda_2\ddot{q}_n\Delta t^2 + \lambda_3\Delta a\Delta t^2\\ \dot{q}_{n+1} &= \dot{q}_n + \lambda_4\ddot{q}_n\Delta t + \lambda_5\Delta a\Delta t\\ \ddot{q}_{n+1} &= \ddot{q}_n + \Delta a\end{aligned} \tag{5.194}$$

Initial conditions:

$$\begin{aligned}q(t_0) &= q_0\\ \dot{q}(t_0) &= \dot{q}_0\end{aligned}$$

Defining $\mathbf{y}_{n+1} := \{q_{n+1}, \Delta t\dot{q}_{n+1}, \Delta t^2\ddot{q}_{n+1}\}^T$ and $\mathbf{y}_n := \{q_n, \Delta t\dot{q}_n, \Delta t^2\ddot{q}_n\}^T$, Eq. (5.194) can be written as

$$\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n + \mathbf{L}_{n+W_1} \tag{5.195}$$

where the amplification matrix $\mathbf{A} \in \mathbb{R}^{3\times3}$ and the external load vector $\mathbf{L}_{n+W_1} \in \mathbb{R}^3$ are designed to be exactly the same form as the single-field form PCE-GSSSS framework of algorithms; see Eq. (5.128) - Eq. (5.130).

Order of Time Accuracy

Previously, the time accuracy analysis in Subsection 5.4.1 concluded that the second-order time accuracy for Eq. (5.232) is achieved under the exact same conditions for the single-field form I-GSSSS framework of algorithms both for the homogeneous and non-homogeneous cases. That is, the necessary and sufficient conditions for the second-order time accuracy are

$$\boxed{\begin{aligned} \lambda_1 &= \lambda_4 = 1 \\ \lambda_2 &= \frac{1}{2} \\ \lambda_5 &= \frac{1}{2}\left[1 + 2W_1(\Lambda_6 - \Lambda_1)\right] \\ \Lambda_1 &= \Lambda_4 \\ \text{All other parameters}&: \text{ arbitrary} \\ &\textbf{(for homogeneous case)} \end{aligned}} \tag{5.196}$$

and

$$\boxed{\begin{aligned} \lambda_1 &= \lambda_4 = 1 \\ \lambda_2 &= \frac{1}{2} \\ \lambda_5 &= \frac{1}{2}\left[1 + 2W_1(\Lambda_6 - 1)\right] \\ \Lambda_1 &= \Lambda_4 = 1 \\ \text{All other parameters}&: \text{ arbitrary} \\ &\textbf{(for non-homogeneous case)} \end{aligned}} \tag{5.197}$$

Spectral Analysis

For the single-field form of either the PCE-GSSSS or E-GSSSS framework of algorithms, it is impossible (not meaningful) to define the principal roots and spurious root at the high-frequency range. In the design of the single-field form PCE-GSSSS framework of algorithms, we directly used the algorithmic parameters in terms of $\rho_{i\infty} \in [0,1]$ $(i = 1,2,3)$ defined originally for the single-field form I-GSSSS framework of algorithms. In the design of the single-field form E-GSSSS framework of algorithms, we instead use the principal root and the spurious root at the bifurcation limit Ω_b, i.e., ρ_b and ρ_{3b}, respectively $(\rho_b \geq \rho_{3b})$, to define the algorithmic parameters. At $\Omega = \Omega_b$, we have the identical real principal roots, i.e., $\rho_b = \rho_{1b} = \rho_{2b} \in \mathbb{R}$, and a real spurious root, $\rho_{3b} \in \mathbb{R}$; hence the characteristic equation for the amplification matrix may be written as

$$(\zeta + \rho_b)^2 (\zeta + \rho_{3b}) = 0 \tag{5.198}$$

Recalling the characteristic polynomial of the amplification matrix is also given from $-\det(\zeta \mathbf{I}_3 - \mathbf{A}) = 0$ as

$$\zeta^3 - A_1^b \zeta^2 + A_2^b \zeta - A_3^b = 0 \tag{5.199}$$

the comparison between Eq. (5.198) and Eq. (5.199) leads to

$$A_1^b := \mathrm{tr}(\mathbf{A}(\Omega_b)) = -(2\rho_b + \rho_{3b}) \tag{5.200}$$

$$A_2^b := \frac{1}{2}\left[(\mathrm{tr}(\mathbf{A}(\Omega_b)))^2 - \mathrm{tr}(\mathbf{A}^2(\Omega_b))\right] = \rho_b^2 + 2\rho_b\rho_{3b} \tag{5.201}$$

$$A_3^b := \det(\mathbf{A}(\Omega_b)) = -\rho_b^2 \tag{5.202}$$

Imposing the conditions for second-order time accuracy given in Eq. (5.196), we can express the algorithmic parameters in terms of ρ_b and ρ_{3b} for the undamped

case ($\xi = 0$) as

$$W_1\Lambda_6 = \frac{2+\rho_{3b}-\rho_{3b}\rho_b}{(1+\rho_b)(1+\rho_{3b})} \tag{5.203}$$

$$W_1\Lambda_1+\lambda_5 = \frac{5+3\rho_{3b}+\rho_b-\rho_{3b}\rho_b}{2(1+\rho_b)(1+\rho_{3b})} \tag{5.204}$$

$$\begin{aligned} W_2\Lambda_2 - W_3\Lambda_3 + \lambda_3 = {} & \frac{\rho_{3b}\rho_b-\rho_{3b}-2}{2\Lambda_6^2(1+\rho_{3b})^2(1+\rho_b)^2}\,[\,2(\rho_{3b}\rho_b-\rho_{3b}-2) \\ & -\Lambda_6\,(\rho_{3b}\rho_b-3\rho_{3b}-\rho_b-5)\,] \\ & -\frac{5+3\rho_b+\rho_{3b}(1-\rho_b)(4+\rho_{3b}+2\rho_b)}{(\rho_{3b}\rho_b-\rho_{3b}-2)(1+\rho_{3b})(1+\rho_b)^2} \end{aligned} \tag{5.205}$$

$$\lambda_1 = \lambda_4 = 1 \tag{5.206}$$

$$\lambda_2 = \frac{1}{2} \tag{5.207}$$

$$\Lambda_1 = \Lambda_4 \tag{5.208}$$

with

$$1 \geq \rho_b \geq \rho_{3b} \geq 0 \tag{5.209}$$

after some tedious work. Note that Λ_5 is a completely independent free parameter. For the non-homogeneous case, we additionally impose $\Lambda_1 = \Lambda_4 = 1$; that is, Eq. (5.204) becomes

$$W_1+\lambda_5 = \frac{5+3\rho_{3b}+\rho_b-\rho_{3b}\rho_b}{2(1+\rho_{3b}+\rho_b+\rho_{3b}\rho_b)} \tag{5.210}$$

For both homogeneous and non-homogeneous undamped cases, the bifurcation limit is given as

$$\Omega_b = \sqrt{(1+\rho_b)(2+\rho_{3b}-\rho_{3b}\rho_b)} \tag{5.211}$$

When selecting $\rho_b = 1$, we have $\Omega_b = \Omega_s \geq 0$ where Ω_s is the stability limit, and it is given as

$$\Omega_b = \Omega_s = 2 \tag{5.212}$$

Time Level Consistency of the Discrete Equation of Motion

We next investigate if the theorem of time level consistency (Theorem 5) still holds for the system represented by Eq. (5.185) - Eq. (5.192). Recall that the

algorithmic time levels of the acceleration vectors are

$$\ddot{\mathbf{q}}_n - \ddot{\mathbf{q}}(t_{n-\phi}) = \mathcal{O}(\Delta t^2) \text{ and } \ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}(t_{n+1-\phi}) = \mathcal{O}(\Delta t^2) \tag{5.213}$$

with $\phi := W_1(\Lambda_6 - 1) \in \mathbb{R}$ according to Theorem 4, whereas the algorithmic time levels of approximations of $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\mathbf{f}$ are simply

$$\begin{aligned}
\mathbf{q}_n - \mathbf{q}(t_n) &= \mathcal{O}(\Delta t^2) \text{ and } \mathbf{q}_{n+1} - \mathbf{q}(t_{n+1}) = \mathcal{O}(\Delta t^2) \\
\dot{\mathbf{q}}_n - \dot{\mathbf{q}}(t_n) &= \mathcal{O}(\Delta t^2) \text{ and } \dot{\mathbf{q}}_{n+1} - \dot{\mathbf{q}}(t_{n+1}) = \mathcal{O}(\Delta t^2) \\
\mathbf{f}_n - \mathbf{f}(t_n) &= \mathcal{O}(\Delta t^2) \text{ and } \mathbf{f}_{n+1} - \mathbf{f}(t_{n+1}) = \mathcal{O}(\Delta t^2)
\end{aligned} \tag{5.214}$$

The algorithmic time level of the algorithmic acceleration vector $\tilde{\mathbf{a}}$ is easily shown as $t = t_{n+W_1}$ from

$$\begin{aligned}
\tilde{\mathbf{a}} &= (1 - W_1\Lambda_6)\ddot{\mathbf{q}}_n + W_1\Lambda_6\ddot{\mathbf{q}}_{n+1} \\
&= (1 - W_1\Lambda_6)\ddot{\mathbf{q}}(t_{n-\phi}) + W_1\Lambda_6\ddot{\mathbf{q}}(t_{n+1-\phi}) + \mathcal{O}(\Delta t^2) \\
&= \ddot{\mathbf{q}}(t_n) + \Delta t W_1 \dddot{\mathbf{q}}(t_n) + \mathcal{O}(\Delta t^2) \\
&= \ddot{\mathbf{q}}(t_{n+W_1})
\end{aligned} \tag{5.215}$$

where we have used the Taylor series expansions of $\ddot{\mathbf{q}}(t_{n-\phi})$ and $\ddot{\mathbf{q}}(t_{n+1-\phi})$ at time t_n. Substituting Eq. (5.191) into Eq. (5.187), the algorithmic time level of the algorithmic velocity is also shown as $t = t_{n+W_1}$ from the following:

$$\begin{aligned}
\tilde{\mathbf{v}} &= \dot{\mathbf{q}}_n + \frac{W_2\Lambda_5}{\lambda_5}(\dot{\mathbf{q}}_{n+1} - \dot{\mathbf{q}}_n) + \Delta t \left[W_1\Lambda_4 - W_2\Lambda_5\frac{\lambda_4}{\lambda_5}\right] \ddot{\mathbf{q}}_n \\
&= \dot{\mathbf{q}}(t_n) + \frac{W_2\Lambda_5}{\lambda_5}\left[\dot{\mathbf{q}}(t_{n+1}) - \dot{\mathbf{q}}(t_n)\right] + \Delta t \left[W_1\Lambda_4 - W_2\Lambda_5\frac{\lambda_4}{\lambda_5}\right] \ddot{\mathbf{q}}(t_n) + \mathcal{O}(\Delta t^2) \\
&= \dot{\mathbf{q}}(t_n) + \Delta t \left[W_1\Lambda_4 + \frac{W_2\Lambda_5}{\lambda_5}(1 - \lambda_4)\right] \ddot{\mathbf{q}}(t_n) + \mathcal{O}(\Delta t^2)
\end{aligned} \tag{5.216}$$

Imposing the second-order time accurate conditions given by Eq. (5.197), Eq. (5.216) yields

$$\tilde{\mathbf{v}} = \dot{\mathbf{q}}(t_n) + \Delta t W_1 \ddot{\mathbf{q}}(t_n) + \mathcal{O}(\Delta t^2) = \dot{\mathbf{q}}(t_{n+W_1}) \tag{5.217}$$

Next, we consider the algorithmic time level of $\check{\mathbf{q}}$. Substituting Eq. (5.190) into Eq. (5.186) and using the Taylor series expansions, we note that algorithmic time level is t_{n+W_1}:

$$\begin{aligned}
\check{\mathbf{q}} &= \mathbf{q}_n + W_1\Lambda_1\dot{\mathbf{q}}_n\Delta t \\
&\quad + \frac{W_2\Lambda_2 - W_3\Lambda_3}{\lambda_2 - \lambda_3}\left[\mathbf{q}_{n+1} - \mathbf{q}_n - \Delta t\dot{\mathbf{q}}_n - \lambda_3\Delta t^2\ddot{\mathbf{q}}_{n+1}\right] \\
&= \mathbf{q}(t_n) + W_1\Lambda_1\dot{\mathbf{q}}(t_n)\Delta t \\
&\quad + \frac{W_2\Lambda_2 - W_3\Lambda_3}{\lambda_2 - \lambda_3}\left[\mathbf{q}(t_{n+1}) - \mathbf{q}(t_n) - \Delta t\dot{\mathbf{q}}(t_n) - \lambda_3\Delta t^2\ddot{\mathbf{q}}(t_{n+1})\right] + \mathcal{O}(\Delta t^2) \\
&= \mathbf{q}(t_n) + W_1\Lambda_1\dot{\mathbf{q}}(t_n)\Delta t \\
&\quad + \frac{W_2\Lambda_2 - W_3\Lambda_3}{\lambda_2 - \lambda_3}\left[\frac{1}{2} - \lambda_3\right]\Delta t^2\ddot{\mathbf{q}}(t_n) + \mathcal{O}(\Delta t^3) \\
&= \mathbf{q}(t_n) + W_1\Lambda_1\dot{\mathbf{q}}(t_n)\Delta t + \mathcal{O}(\Delta t^2) \\
&= \mathbf{q}(t_{n+W_1}) \qquad (5.218)
\end{aligned}$$

due to $\Lambda_1 = 1$; see Eq. (5.197). Therefore, it is clear that Theorem 5 still holds for the single-field form E-GSSSS framework of algorithms with the implicit treatment of the velocity term. That is,

$$\mathbf{0} = \mathbf{M}\tilde{\mathbf{a}} + \mathbf{C}\tilde{\mathbf{v}} + \mathbf{K}\check{\mathbf{q}} - \tilde{\mathbf{f}} + \mathcal{O}(\Delta t^2) = \mathbf{M}\ddot{\mathbf{q}}(t^*) + \mathbf{C}\dot{\mathbf{q}}(t^*) + \mathbf{K}\mathbf{q}(t^*) - \mathbf{f}(t^*) \quad (5.219)$$

where the consistent algorithmic time level is $t^* = t_{n+W_1}$. ***The single-field form E-GSSSS framework of algorithms with the implicit treatment of the velocity term has the same consistent algorithmic time level as the single-field form I-GSSSS framework of algorithms.***

Algorithm 10

Single-field Form E-GSSSS Framework of Algorithms for Linear Dynamical Systems: Implicit Treatment of the Velocity Term

Integrator:

$$\left[\mathbf{M}W_1\Lambda_6 + \mathbf{C}W_2\Lambda_5\Delta t\right]\Delta\mathbf{a} = \tilde{\mathbf{f}} - \mathbf{K}\check{\mathbf{q}} - \mathbf{C}\left[\dot{\mathbf{q}}_n + W_1\Lambda_4\ddot{\mathbf{q}}_n\Delta t\right] - \mathbf{M}\ddot{\mathbf{q}}_n$$

where

$$\begin{aligned} \check{\mathbf{q}} &= \mathbf{q}_n + W_1\Lambda_1\dot{\mathbf{q}}_n\Delta t + (W_2\Lambda_2 - W_3\Lambda_3)\ddot{\mathbf{q}}_n\Delta t^2 \\ \tilde{\mathbf{f}} &= \mathbf{f}_n + W_1(\mathbf{f}_{n+1} - \mathbf{f}_n) \;\; or \;\; \mathbf{f}(t_{n+W_1}) \end{aligned} \tag{5.220}$$

Updates:

$$\begin{aligned} \mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + \lambda_3\Delta\mathbf{a}\Delta t^2 \\ \dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5\Delta\mathbf{a}\Delta t \\ \ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_n + \Delta\mathbf{a} \end{aligned} \tag{5.221}$$

Initial conditions:

$$\begin{aligned} \mathbf{q}(t_0) &= \mathbf{q}_0 \\ \boldsymbol{\nu}(t_0) &= \boldsymbol{\nu}_0 \end{aligned}$$

Algorithmic parameters (for second-order time accuracy):

$$\begin{aligned} W_1\Lambda_6 &= \frac{2 + \rho_{3b} - \rho_{3b}\rho_b}{(1+\rho_b)(1+\rho_{3b})} \\ W_1\Lambda_1 + \lambda_5 &= \frac{5 + 3\rho_{3b} + \rho_b - \rho_{3b}\rho_b}{2(1+\rho_b)(1+\rho_{3b})} \\ W_2\Lambda_2 - W_3\Lambda_3 + \lambda_3 &= \frac{\rho_{3b}\rho_b - \rho_{3b} - 2}{2\Lambda_6^2(1+\rho_{3b})^2(1+\rho_b)^2}\,[\,2(\rho_{3b}\rho_b - \rho_{3b} - 2) \\ &\quad - \Lambda_6\,(\rho_{3b}\rho_b - 3\rho_{3b} - \rho_b - 5)\,] \\ &\quad - \frac{5 + 3\rho_b + \rho_{3b}(1-\rho_b)(4 + \rho_{3b} + 2\rho_b)}{(\rho_{3b}\rho_b - \rho_{3b} - 2)(1+\rho_{3b})(1+\rho_b)^2} \\ \lambda_1 &= \lambda_4 = 1 \\ \lambda_2 &= \frac{1}{2} \\ \Lambda_1 &= \Lambda_4 \end{aligned}$$

It is important to note that some free parameters remain in the algorithm shown above; however, regardless the choices of the free parameters, the second-order time accuracy of the algorithm in the configuration, velocity, and acceleration vectors can be achieved. The famous ***central difference method*** (explicit

Newmark method with $\gamma = 1/2$), which is equivalent to the so-called ***velocity verlet algorithm*** [41], in the sense of the implicit treatment of the velocity term, can be obtained by selecting $(\rho_b, \rho_{3b}) = (1, 0)$ as expected. Recall that the same scheme (central difference method/velocity verlet algorithm) can be also obtained by selecting $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}}) = (1, 1, 0)$ in Algorithm 6.

5.5.2 Explicit Treatment of the Velocity Term

From the single-field form I-GSSSS framework of algorithms, we propose the representation of the single-field form E-GSSSS algorithmic structure with the explicit treatment of the velocity term as follows:

$$\mathbf{M}\tilde{\mathbf{a}} + \mathbf{C}\check{\mathbf{v}} + \mathbf{K}\check{\mathbf{q}} = \tilde{\mathbf{f}} \tag{5.222}$$

where

$$\check{\mathbf{q}} = \mathbf{q}_n + W_1\Lambda_1\dot{\mathbf{q}}_n\Delta t + (W_2\Lambda_2 - W_3\Lambda_3)\ddot{\mathbf{q}}_n\Delta t^2 \tag{5.223}$$

$$\check{\mathbf{v}} = \dot{\mathbf{q}}_n + (W_1\Lambda_4 - W_2\Lambda_5)\ddot{\mathbf{q}}_n\Delta t \tag{5.224}$$

$$\tilde{\mathbf{a}} = \ddot{\mathbf{q}}_n + W_1\Lambda_6\Delta\mathbf{a} \tag{5.225}$$

$$\tilde{\mathbf{f}} = \mathbf{f}_n + W_1(\mathbf{f}_{n+1} - \mathbf{f}_n) \ \text{ or } \ \mathbf{f}(t_{n+W_1}) \tag{5.226}$$

And the associated updates are:

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + \lambda_3\Delta\mathbf{a}\Delta t^2 \tag{5.227}$$

$$\dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5\Delta\mathbf{a}\Delta t \tag{5.228}$$

$$\ddot{\mathbf{q}}_{n+1} = \ddot{\mathbf{q}}_n + \Delta\mathbf{a} \tag{5.229}$$

The above expression is in the A-form (likewise the V- and D-forms may be constructed); that is, we find $\Delta\mathbf{a}$ as the primary unknown from the balance equation

$$W_1\Lambda_6\mathbf{M}\Delta\mathbf{a} = \tilde{\mathbf{f}} - \mathbf{M}\ddot{\mathbf{q}}_n - \mathbf{C}\check{\mathbf{v}} - \mathbf{K}\check{\mathbf{q}} \tag{5.230}$$

and then compute $\mathbf{q}_{n+1}$, $\dot{\mathbf{q}}_{n+1}$, and $\ddot{\mathbf{q}}_{n+1}$ for $n \in \{0, 1, 2, \cdots, N\}$ with given initial conditions, $\mathbf{q}(t_0) = \mathbf{q}_0$ and $\dot{\mathbf{q}}(t_0) = \dot{\mathbf{q}}_0$. The modal decomposition for the new

explicit framework yields

$$
\begin{aligned}
&\textbf{\textit{Integrator:}}\\
&W_1\Lambda_6\Delta a\\
&= \tilde{g} - \ddot{q}_n - 2\xi\omega\left[\dot{q}_n + (W_1\Lambda_4 - W_2\Lambda_5)\ddot{q}_n\Delta t\right]\\
&- \omega^2\left[q_n + W_1\Lambda_1\dot{q}_n\Delta t + (W_2\Lambda_2 - W_3\Lambda_3)\ddot{q}_n\Delta t^2\right]\\
&\textbf{\textit{Updates:}}\\
&q_{n+1} = q_n + \lambda_1\dot{q}_n\Delta t + \lambda_2\ddot{q}_n\Delta t^2 + \lambda_3\Delta a\Delta t^2 \qquad (5.231)\\
&\dot{q}_{n+1} = \dot{q}_n + \lambda_4\ddot{q}_n\Delta t + \lambda_5\Delta a\Delta t\\
&\ddot{q}_{n+1} = \ddot{q}_n + \Delta a\\
&\textbf{\textit{Initial conditions:}}\\
&q(t_0) = q_0\\
&\dot{q}(t_0) = \dot{q}_0
\end{aligned}
$$

Defining $\mathbf{y}_{n+1} := \{q_{n+1}, \Delta t\dot{q}_{n+1}, \Delta t^2\ddot{q}_{n+1}\}^T$ and $\mathbf{y}_n := \{q_n, \Delta t\dot{q}_n, \Delta t^2\ddot{q}_n\}^T$, Eq. (5.231) can be written as

$$\mathbf{y}_{n+1} = \mathbf{A}\mathbf{y}_n + \mathbf{L}_{n+W_1} \qquad (5.232)$$

where the amplification matrix $\mathbf{A} \in \mathbb{R}^{3\times3}$ and the external load vector $\mathbf{L}_{n+W_1} \in \mathbb{R}^3$ are designed to be exactly the same form for the single-field form PCE-GSSSS framework of algorithms; see Eq. (5.167) - Eq. (5.169).

Order of Time Accuracy

Since the amplification matrix for the system constituted by Eq. (5.231) is identical to the amplification matrix for the single-field form PCE-GSSSS framework of algorithms with the explicit treatment of the velocity term, the necessary and sufficient conditions for the second-order time accuracy are readily given as follows;

see Subsection 5.4.2 for details:

$$\boxed{\begin{aligned} \lambda_1 &= \lambda_4 = 1 \\ \lambda_2 &= \frac{1}{2} \\ \lambda_5 &= \frac{1}{2}\left[1 + 2W_1(\Lambda_6 - \Lambda_1)\right] \\ W_1(\Lambda_4 - \Lambda_1) &= W_2\Lambda_5 \\ \text{All other parameters}: &\quad \text{arbitrary} \\ &\textbf{(for homogeneous case)} \end{aligned}} \tag{5.233}$$

and

$$\boxed{\begin{aligned} \lambda_1 &= \lambda_4 = 1 \\ \lambda_2 &= \frac{1}{2} \\ \lambda_5 &= \frac{1}{2}\left[1 + 2W_1(\Lambda_6 - 1)\right] \\ W_1(\Lambda_4 - 1) &= W_2\Lambda_5 \\ \Lambda_1 &= 1 \\ \text{All other parameters}: &\quad \text{arbitrary} \\ &\textbf{(for non-homogeneous case)} \end{aligned}} \tag{5.234}$$

Spectral Analysis

Following a similar procedure in Subsection 5.5.1, we relate the algorithmic parameters with the principal root and the spurious root at the bifurcation limit Ω_b, i.e., ρ_b and ρ_{3b}, respectively ($\rho_b \geq \rho_{3b}$). At $\Omega = \Omega_b$, we have the identical real principal roots, i.e., $\rho_b = \rho_{1b} = \rho_{2b} \in \mathbb{R}$, and real spurious root, $\rho_{3b} \in \mathbb{R}$. Therefore, the comparison between

$$(\zeta + \rho_b)^2 (\zeta + \rho_{3b}) = 0 \tag{5.235}$$

and the characteristic polynomial of the amplification matrix

$$\zeta^3 - A_1^b\zeta^2 + A_2^b\zeta - A_3^b = 0 \tag{5.236}$$

leads to

$$A_1^b := \mathrm{tr}(\mathbf{A}(\Omega_b)) = -(2\rho_b + \rho_{3b}) \tag{5.237}$$

$$A_2^b := \frac{1}{2}\left[(\mathrm{tr}(\mathbf{A}(\Omega_b)))^2 - \mathrm{tr}(\mathbf{A}^2(\Omega_b))\right] = \rho_b^2 + 2\rho_b\rho_{3b} \tag{5.238}$$

$$A_3^b := \det(\mathbf{A}(\Omega_b)) = -\rho_b^2 \tag{5.239}$$

Imposing the conditions for the second-order time accuracy given in Eq. (5.233), we can express the algorithmic parameters in terms of ρ_b and ρ_{3b} for the undamped case ($\xi = 0$) as

$$W_1\Lambda_6 = \frac{2 + \rho_{3b} - \rho_{3b}\rho_b}{(1+\rho_b)(1+\rho_{3b})} \tag{5.240}$$

$$W_1\Lambda_1 + \lambda_5 = \frac{5 + 3\rho_{3b} + \rho_b - \rho_{3b}\rho_b}{2(1+\rho_b)(1+\rho_{3b})} \tag{5.241}$$

$$\begin{aligned} W_2\Lambda_2 - W_3\Lambda_3 + \lambda_3 = {} & \frac{\rho_{3b}\rho_b - \rho_{3b} - 2}{2\Lambda_6^2(1+\rho_{3b})^2(1+\rho_b)^2}\,[\,2(\rho_{3b}\rho_b - \rho_{3b} - 2) \\ & -\Lambda_6\,(\rho_{3b}\rho_b - 3\rho_{3b} - \rho_b - 5)\,] \\ & -\frac{5 + 3\rho_b + \rho_{3b}(1-\rho_b)(4 + \rho_{3b} + 2\rho_b)}{(\rho_{3b}\rho_b - \rho_{3b} - 2)(1+\rho_{3b})(1+\rho_b)^2} \end{aligned} \tag{5.242}$$

$$\lambda_1 = \lambda_4 = 1 \tag{5.243}$$

$$\lambda_2 = \frac{1}{2} \tag{5.244}$$

$$W_2\Lambda_5 = W_1(\Lambda_4 - \Lambda_1) \tag{5.245}$$

with

$$1 \geq \rho_b \geq \rho_{3b} \geq 0 \tag{5.246}$$

For the non-homogeneous case, additionally impose $\Lambda_1 = 1$; that is, Eq. (5.241) becomes

$$W_1 + \lambda_5 = \frac{5 + 3\rho_{3b} + \rho_b - \rho_{3b}\rho_b}{2(1 + \rho_{3b} + \rho_b + \rho_{3b}\rho_b)} \tag{5.247}$$

Substituting Eq. (5.241) into Eq. (5.245), we can eliminate $W_1\Lambda_1$ as

$$W_1\Lambda_4 - W_2\Lambda_5 + \lambda_5 = \frac{5 + 3\rho_{3b} + \rho_b - \rho_{3b}\rho_b}{2(1+\rho_b)(1+\rho_{3b})} \tag{5.248}$$

For both homogeneous and non-homogeneous undamped cases, the bifurcation limit is given as

$$\Omega_b = \sqrt{(1+\rho_b)(2+\rho_{3b}-\rho_{3b}\rho_b)} \tag{5.249}$$

which is the same for the case of the implicit treatment of the velocity term.

Time Level Consistency of the Discrete Equation of Motion

Again, let us investigate if the theorem of the time level consistency (Theorem 5) still holds for the system represented by Eq. (5.222) - Eq. (5.229). Recall that we have

$$\begin{aligned} \tilde{\mathbf{a}} &= \ddot{\mathbf{q}}(t_n) + W_1 \dddot{\mathbf{q}}(t_n)\Delta t + \mathcal{O}(\Delta t^2) = \ddot{\mathbf{q}}(t_{n+W_1}) && (5.250)\\ \check{\mathbf{q}} &= \mathbf{q}(t_n) + W_1 \dot{\mathbf{q}}(t_n)\Delta t + \mathcal{O}(\Delta t^2) = \mathbf{q}(t_{n+W_1}) && (5.251) \end{aligned}$$

from the previous subsection. For the single-field form E-GSSSS framework of algorithms with the explicit treatment of the velocity term, we must show the algorithmic time level of $\check{\mathbf{v}}$ instead of $\tilde{\mathbf{v}}$. Substituting Eq. (5.228) into Eq. (5.224) with the condition from Eq. (5.234),

$$\begin{aligned} \lambda_5 &= \frac{1}{2} + \phi \\ W_1 &= W_1\Lambda_4 - W_2\Lambda_5 \end{aligned} \tag{5.252}$$

where $\phi = W_1(\Lambda_6 - 1)$, we get

$$\begin{aligned} \check{\mathbf{v}} &= \dot{\mathbf{q}}_n + \frac{2W_1}{1-2\phi}\left[\dot{\mathbf{q}}_{n+1} - \dot{\mathbf{q}}_n\right] - W_1\frac{1+2\phi}{1-2\phi}\Delta t\ddot{\mathbf{q}}_{n+1} \\ &= \dot{\mathbf{q}}(t_n) + \frac{2W_1}{1-2\phi}\left[\dot{\mathbf{q}}(t_{n+1}) - \dot{\mathbf{q}}(t_n)\right] - W_1\frac{1+2\phi}{1-2\phi}\Delta t\ddot{\mathbf{q}}(t_{n+1-\phi}) + \mathcal{O}(\Delta t^2) \\ &= \dot{\mathbf{q}}(t_n) + W_1\ddot{\mathbf{q}}(t_n)\Delta t + \mathcal{O}(\Delta t^2) \\ &= \dot{\mathbf{q}}(t_{n+W_1}) \end{aligned} \tag{5.253}$$

Therefore, the algorithmic time level of $\check{\mathbf{v}}$ is also t_{n+W_1}. Hence, Theorem 5 still holds for the single-field form E-GSSSS framework of algorithms with the explicit treatment of the velocity term:

$$\mathbf{0} = \mathbf{M}\tilde{\mathbf{a}} + \mathbf{C}\check{\mathbf{v}} + \mathbf{K}\check{\mathbf{q}} - \tilde{\mathbf{f}} + \mathcal{O}(\Delta t^2) = \mathbf{M}\ddot{\mathbf{q}}(t^*) + \mathbf{C}\dot{\mathbf{q}}(t^*) + \mathbf{K}\mathbf{q}(t^*) - \mathbf{f}(t^*) \tag{5.254}$$

where the consistent algorithmic time level is $t^* = t_{n+W_1}$. ***The single-field form E-GSSSS framework of algorithms with the explicit treatment of the velocity term has the same consistent algorithmic time level as the single-field form I-GSSSS framework of algorithms.***

Algorithm 11

Single-field Form E-GSSSS Framework of Algorithms for Linear Dynamical Systems: Explicit Treatment of the Velocity Term

Integrator:

$$W_1\Lambda_6\mathbf{M}\Delta\mathbf{a} = \tilde{\mathbf{f}} - \mathbf{K}\check{\mathbf{q}} - \mathbf{C}\check{\mathbf{v}} - \mathbf{M}\ddot{\mathbf{q}}_n$$

where

$$\begin{aligned}
\check{\mathbf{q}} &= \mathbf{q}_n + W_1\Lambda_1\dot{\mathbf{q}}_n\Delta t + (W_2\Lambda_2 - W_3\Lambda_3)\ddot{\mathbf{q}}_n\Delta t^2 \\
\check{\mathbf{v}} &= \dot{\mathbf{q}}_n + (W_1\Lambda_4 - W_2\Lambda_5)\ddot{\mathbf{q}}_n\Delta t \\
\tilde{\mathbf{f}} &= \mathbf{f}_n + W_1(\mathbf{f}_{n+1} - \mathbf{f}_n) \;\; or \;\; \mathbf{f}(t_{n+W_1})
\end{aligned} \tag{5.255}$$

Updates:

$$\begin{aligned}
\mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + \lambda_3\Delta\mathbf{a}\Delta t^2 \\
\dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5\Delta\mathbf{a}\Delta t \\
\ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_n + \Delta\mathbf{a}
\end{aligned} \tag{5.256}$$

Initial conditions:

$$\begin{aligned}
\mathbf{q}(t_0) &= \mathbf{q}_0 \\
\boldsymbol{\nu}(t_0) &= \boldsymbol{\nu}_0
\end{aligned}$$

Algorithmic parameters (for second-order time accuracy):

$$
\begin{aligned}
W_1\Lambda_6 &= \frac{2+\rho_{3b}-\rho_{3b}\rho_b}{(1+\rho_b)(1+\rho_{3b})} \\
W_1\Lambda_1+\lambda_5 &= \frac{5+3\rho_{3b}+\rho_b-\rho_{3b}\rho_b}{2(1+\rho_b)(1+\rho_{3b})} \\
W_2\Lambda_2-W_3\lambda_3+\lambda_3 &= \frac{\rho_{3b}\rho_b-\rho_{3b}-2}{2\Lambda_6^2(1+\rho_{3b})^2(1+\rho_b)^2}\,[\,2(\rho_{3b}\rho_b-\rho_{3b}-2) \\
&\quad -\Lambda_6\,(\rho_{3b}\rho_b-3\rho_{3b}-\rho_b-5)\,] \\
&\quad -\frac{5+3\rho_b+\rho_{3b}(1-\rho_b)(4+\rho_{3b}+2\rho_b)}{(\rho_{3b}\rho_b-\rho_{3b}-2)(1+\rho_{3b})(1+\rho_b)^2} \\
\lambda_1 &= \lambda_4 = 1 \\
\lambda_2 &= \frac{1}{2} \\
W_2\Lambda_5 &= W_1(\Lambda_4-\Lambda_1)
\end{aligned}
$$

It is important to note that some free parameters remain in the algorithm shown above; however, regardless the choices of the free parameters, the second-order time accuracy of the algorithm in the configuration, velocity, and acceleration vectors can be achieved. The famous ***central difference method*** (explicit Newmark method with $\gamma = 1/2$), which is equivalent to the so-called ***velocity verlet algorithm*** [41], in the sense of the explicit treatment of the velocity term, can be obtained by selecting $(\rho_b, \rho_{3b}) = (1, 0)$ as expected. Recall that the same scheme (central difference method/velocity verlet algorithm) can be also obtained by selecting $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}}) = (1, 1, 0)$ in Algorithm 8.

5.6 Unified Algorithmic Framework in the Single-field Form

In view of the implicit GSSSS framework of algorithms in the sense of the single-field form in the second-order system, i.e., the three-root system with the two principal roots and one spurious root, $\rho_\infty^{\max}$, $\rho_\infty^{\min}$, and ρ_∞^{s}, we can unify the general algorithmic forms without loss of generality in the following single algorithm

package via the following replacements,

$$\begin{aligned} \Lambda_5 &\rightarrow \Lambda_5\eta_1 \\ \Lambda_3 &\rightarrow \Lambda_3\eta_2 \end{aligned} \tag{5.257}$$

where η_1 and η_2 are the additional algorithmic parameters. For the update of the configuration, the following replacement

$$\lambda_3 \rightarrow \lambda_3\eta_3 \tag{5.258}$$

is employed.

Algorithm 12

Unified Representation: Single-Field Form GSSSS Framework of Algorithms for Linear Dynamical Systems

Integrator:

$$\mathbf{M}\tilde{\mathbf{a}} + \mathbf{C}\tilde{\mathbf{v}} + \mathbf{K}\tilde{\mathbf{q}} = \tilde{\mathbf{f}}$$

where

$$\begin{aligned} \tilde{\mathbf{q}} &= \mathbf{q}_n + W_1\Lambda_1\dot{\mathbf{q}}_n\Delta t + W_2\Lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + W_3\Lambda_3\eta_2\Delta\mathbf{a}\Delta t^2 \\ \tilde{\mathbf{v}} &= \dot{\mathbf{q}}_n + W_1\Lambda_4\ddot{\mathbf{q}}_n\Delta t + W_2\Lambda_5\eta_1\Delta\mathbf{a}\Delta t \\ \tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1\Lambda_6\Delta\mathbf{a} \\ \tilde{\mathbf{f}} &= \mathbf{f}_n + W_1(\mathbf{f}_{n+1} - \mathbf{f}_n) \;\; or \;\; \mathbf{f}(t_{n+W_1}) \end{aligned}$$

And the associated updates are:

$$\begin{aligned} \mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + \lambda_3\eta_3\Delta\mathbf{a}\Delta t^2 \\ \dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5\Delta\mathbf{a}\Delta t \\ \ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_n + \Delta\mathbf{a} \end{aligned}$$

Initial conditions:

$$\begin{aligned} \mathbf{q}(t_0) &= \mathbf{q}_0 \\ \dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0 \end{aligned}$$

Remark 13 (Algorithm 12)

1. ***I-GSSSS Framework of Algorithms:*** *When $\eta_1 = \eta_2 = \eta_3 = 1$, Algorithm 12 recovers the general implicit algorithmic structures of the U0 and V0 families of algorithms, i.e., Algorithm 2 and Algorithm 3, respectively.*

2. ***E-GSSSS Framework of Algorithms (Implicit Treatment of the Velocity Term):*** *When $\eta_1 = 1$ and $\eta_2 = 0$, Algorithm 12 recovers the general explicit algorithmic structures with the implicit treatment of the velocity term. For the U0 and V0 families of PCE-GSSSS framework of algorithms, we employ the algorithmic parameters from Algorithm 2 and Algorithm 3, respectively. To recover Algorithm 10, we employ the algorithmic parameters defined as*

$$\begin{aligned}
W_1\Lambda_6 &= \frac{2+\rho_{3b}-\rho_{3b}\rho_b}{(1+\rho_b)(1+\rho_{3b})} \\
W_1\Lambda_1+\lambda_5 &= \frac{5+3\rho_{3b}+\rho_b-\rho_{3b}\rho_b}{2(1+\rho_b)(1+\rho_{3b})} \\
W_2\Lambda_2+\lambda_3 &= \frac{\rho_{3b}\rho_b-\rho_{3b}-2}{2\Lambda_6^2(1+\rho_{3b})^2(1+\rho_b)^2}\,[\,2(\rho_{3b}\rho_b-\rho_{3b}-2) \\
&\quad -\Lambda_6\,(\rho_{3b}\rho_b-3\rho_{3b}-\rho_b-5)\,] \\
&\quad -\frac{5+3\rho_b+\rho_{3b}(1-\rho_b)(4+\rho_{3b}+2\rho_b)}{(\rho_{3b}\rho_b-\rho_{3b}-2)(1+\rho_{3b})(1+\rho_b)^2} \\
\lambda_1 &= \lambda_4 = 1 \\
\lambda_2 &= \frac{1}{2} \\
\Lambda_1 &= \Lambda_4
\end{aligned} \tag{5.259}$$

for the undamped case.

3. ***E-GSSSS Framework of Algorithms (Explicit Treatment of the Velocity Term):*** *When $\eta_1 = \eta_2 = 0$, Algorithm 12 recovers the general explicit algorithmic structure with the explicit treatment of the velocity term. For the U0 and V0 families of PCE-GSSSS framework of algorithms, we*

employ the algorithmic parameters from Algorithm 2 and Algorithm 3, respectively. To recover Algorithm 11, we employ the algorithmic parameters defined in Eq. (5.259) *for the undamped case.*

4. *Typical common explicit schemes which can be recovered are summarized in Table 5.1; many of the others are new.*

Algorithms	Conditions
Central Difference/Velocity Verlet	U0($\rho_\infty^{\min}$,$\rho_\infty^{\max}$,ρ_∞^{s},η_1,η_2,η_3)=(1,1,0,1,0,0) - PCE
	(ρ_{3b},ρ_b,η_1,η_2,η_3)=(1,1,1,0,0)
Explicit Generalized-α [42]	(ρ_{3b},ρ_b,η_1,η_2,η_3)=(ρ,ρ,1,0,0)

Table 5.1: Common explicit schemes in Algorithm 12 - Single-field form

5.7 Numerical Illustrations of the Family of the Explicit GSSSS Algorithms - Single-field Form

The plots of the time accuracy, stability, mechanical energy, numerical dissipation, and numerical dispersion of selected schemes within the family of the PCE-GSSSS algorithms are shown in this section. The discussions and summaries of the Family of the Explicit GSSSS Algorithms in the single-field form are as follows. Since PCE and general explicit families of GSSSS algorithms share the important schemes, we show the numerical results only from the sense of PCE-GSSSS family of algorithms.

Time Accuracy: The order of the time accuracy of the PCE-GSSSS family of algorithms with either implicit or explicit treatment of the velocity term (IT or ET), in the linear, dissipative (damped), forced systems, is second-order regardless of any choices for the algorithmic parameters.

Selected Algorithms:

$$\begin{aligned}
\text{U0V0/V0U0}(1.0, 1.0, 1.0): &\ \text{MPR-EPA Based} \\
\text{U0V0}(1.0, 1.0, 0.0): &\ \text{Implicit Newmark Based} \\
\text{V0U0}(1.0, 1.0, 0.0): &\ \text{MPR-MPA Based} \\
\text{U0V0/V0U0}(0.8, 1.0, 0.8): &\ \text{U0V0/V0U0 Optimal Based} \\
\text{U0V1}(0.6, 0.9, 0.2) & \\
\text{V0U1}(0.6, 0.9, 0.2) &
\end{aligned}$$

Fig. 5.1 - Fig. 5.4 shows the order of time accuracy of the configuration ($\square$), velocity ($\triangle$), and acceleration ($\circ$) for the selected schemes with implicit treatment (IT) and explicit treatment (ET), respectively. The configuration, velocity, and acceleration have been used to calculate the reference errors,

$$\begin{aligned}
\text{error}_q &= \left| \frac{q - q_{\text{ref}}}{q_{\text{ref}}} \right| \\
\text{error}_v &= \left| \frac{\dot{q} - \dot{q}_{\text{ref}}}{\dot{q}_{\text{ref}}} \right| \\
\text{error}_a &= \left| \frac{\ddot{q} - \ddot{q}_{\text{ref}}}{\ddot{q}_{\text{ref}}} \right|
\end{aligned} \tag{5.260}$$

respectively, at time $t = 10$ sec. The reference configurations, velocities, and accelerations have been obtained directly from the exact solutions, i.e., $q_{\text{ref}} = q(10)$, $\dot{q}_{\text{ref}} = \dot{q}(10)$, and $\ddot{q}_{\text{ref}} = \ddot{q}(10)$, respectively, or they can be obtained with a sufficiently small time step size $\Delta t = 10^{-5}$ or lower. The following time step sizes were employed for q, $\dot{q}$, and $\ddot{q}$: $\Delta t_1 = 10^{-2}$, $\Delta t_2 = 10^{-3}$, and $\Delta t_3 = 10^{-4}$. We have selected $\eta_3 = 1$ for Fig. 5.1 and Fig. 5.2 whereas $\eta_3 = 0$ for Fig. 5.3 and Fig. 5.4. The velocity term in the equation, $\ddot{q} + \dot{q} + 10q = 0.5\sin(t)$ with $q_0 = 1$ and $\dot{q}_0 = 1$, has been implicitly treated ($\eta_1 = 1$) for Fig. 5.1 and Fig. 5.3 whereas it has been explicitly treated $\eta_1 = 0$ for Fig. 5.2 and Fig. 5.4. As can be seen from the figures, all time integration schemes are second-order time accurate in all the configurations, velocities, and accelerations.

Stability (Spectral Plot): The spectral plots of some selected schemes from the family of the PCE-GSSSS algorithms in the conservative system are shown in Fig. 5.5 ($\eta_3 = 1$) and Fig. 5.6 ($\eta_3 = 0$). In the figures, $\rho_\infty^{\min}$, $\rho_\infty^{\max}$, and ρ_∞^{s} are denoted by blue-colored curves with $\square$, red-colored curves with $\triangle$, and black-colored curves with $\circ$, respectively. It should be noted that all schemes are conditionally stable, and U0V0/V0U0$(1, 1, 1)$ scheme with $\eta_3 = 1$ is unconditionally unstable as expected from the theory. The selected explicit schemes for the stability analysis (in Fig. 5.5 and Fig. 5.6) included here are:

U0V0/V0U0$(1.0, 1.0, 1.0)$: MPR-EPA Based
U0V0/V0U0$(0.8, 1.0, 0.8)$: U0V0/V0U0 Optimal Based
U0V0$(1.0, 1.0, 0.0)$: Implicit Newmark Based
V0U0$(1.0, 1.0, 0.0)$: MPR-MPA Based
U0V1$(0.8, 0.8, 0.8)$: Three Parameter Optimal Based
V0U1$(0.8, 0.8, 0.8)$: V0 Counterpart of Three Parameter Optimal Based
U0V1$(0.0, 0.0, 0.0)$: L-Stable Three Parameter Optimal Based
V0U1$(0.0, 0.0, 0.0)$: V0 Counterpart of L-Stable Three Parameter Optimal Based
U0V1$(0.8, 0.8, 0.0)$: WBZ Based
V0U1$(0.8, 0.8, 0.0)$: V0 Counterpart of WBZ Based
U0V1$(0.8, 0.9, 0.09)$: HHT-α Based
V0U1$(0.8, 0.9, 0.09)$: V0 Counterpart of HHT-α Based

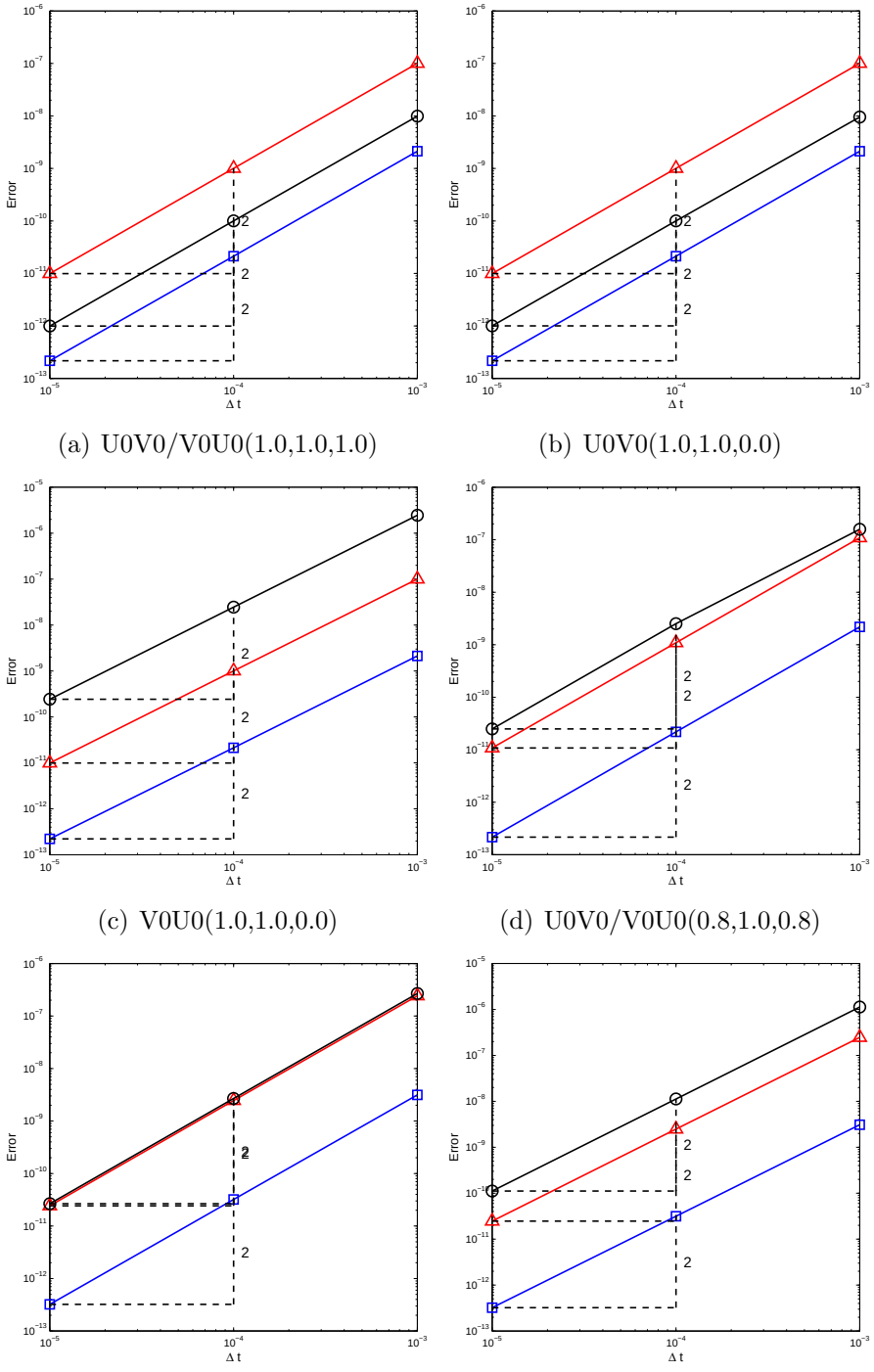
Fig. 5.7 shows a comparison of the behavior of the spectral radius, i.e., $\rho_\infty^{\max}$ for the cases where $\eta_3 = 1$ and $\eta_3 = 0$. It should be noted that the critical time step can never achieve 2 if we select $\eta_3 = 1$; however, by selecting $\eta_3 = 0$ it tends to make the schemes more stable. It appears that when we introduce the numerical dissipation into the schemes, then it makes the critical time step of these schemes lower than the numerically nondissipative schemes.

Mechanical Energy: The time histories of the mechanical energy of the conservative system are shown in Fig. 5.8 ($\eta_3 = 1$) and Fig. 5.9 ($\eta_3 = 0$) for selected explicit schemes. No member can conserve the mechanical energy exactly. Whereas all schemes in Fig. 5.8 show the dissipation in the mechanical energy,

the following schemes in Fig. 5.9 show that the mechanical energy is bounded: U0V0/V0U0(1,1,1), U0V0/V0U0(0,1,0), U0V0(1,1,ρ_∞^{s}), U0V0($\rho_\infty^{\min}$,1,0), and U0V1(0,0,0). The enlarged legends of the figures are shown in Fig. 4.5.

Numerical Dissipation and Numerical Dispersion: The numerical dissipation and numerical dispersion plots of selected schemes in the family of the PCE-GSSSS algorithms in the conservative system are shown in Fig. 5.10 ($\eta_3 = 1$) and Fig. 5.11 ($\eta_3 = 0$). Similar to the implicit case in the previous chapter, there does not exist any scheme which has zero numerical dispersion regardless of the choice of η_3. However, it is noteworthy that there exists some schemes with no numerical dissipation when we select $\eta_3 = 0$. Although there exists no scheme of zero numerical dissipation in Fig. 5.10, the following schemes, U0V0/V0U0(1,1,1), U0V0/V0U0(0,1,0), U0V0(1,1,ρ_∞^{s}), U0V0($\rho_\infty^{\min}$,1,0), and U0V1(0,0,0), in Fig. 5.11 show zero numerical dissipation features. The enlarged legends of the figures are shown in Fig. 4.5.

The numerical results show that the second-order time accurate, conditionally stable framework of explicit GSSSS algorithms include the variational and exact momentum conserving family of schemes in the linear dynamical conservative systems. By turning off the numerical dissipation features, we can recover the numerically non-dissipative family of schemes.



(a) U0V0/V0U0(1.0,1.0,1.0)

(b) U0V0(1.0,1.0,0.0)

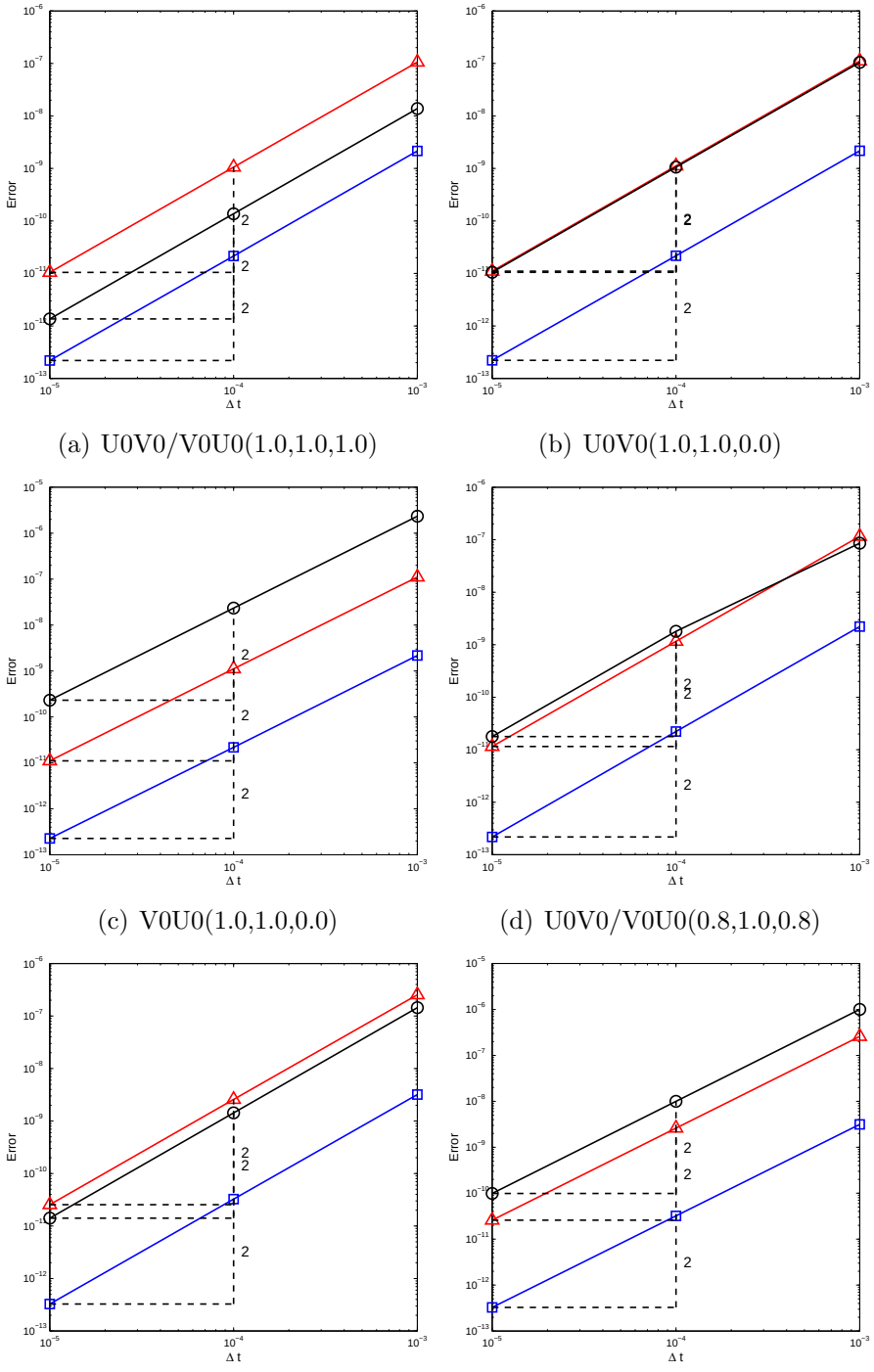
(c) V0U0(1.0,1.0,0.0)

(d) U0V0/V0U0(0.8,1.0,0.8)

(e) U0V1(0.6,0.9,0.2)

(f) V0U1(0.6,0.9,0.2)

Figure 5.1: Time Accuracy plots in the configuration (□), velocity (△), and acceleration (○) within the PCE-IT GSSSS family of algorithms with $\eta_3 = 1$ for the linear dissipative nonhomogeneous system, $\ddot{q} + \dot{q} + 10q = 0.5\sin(t)$, with the initial conditions $q_0 = 1$ and $\dot{q}_0 = 1$ - Single-field form



(a) U0V0/V0U0(1.0,1.0,1.0)

(b) U0V0(1.0,1.0,0.0)

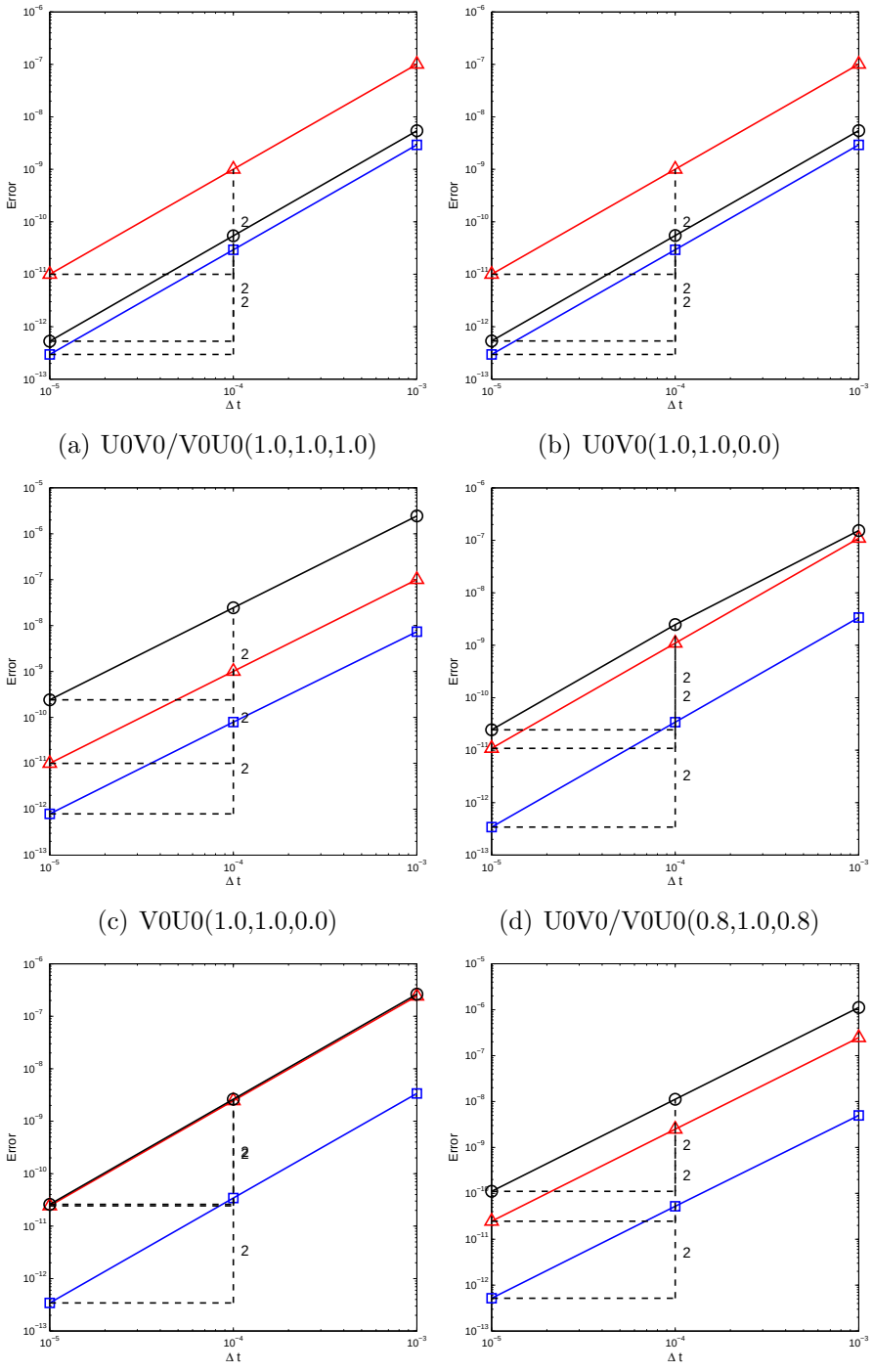
(c) V0U0(1.0,1.0,0.0)

(d) U0V0/V0U0(0.8,1.0,0.8)

(e) U0V1(0.6,0.9,0.2)

(f) V0U1(0.6,0.9,0.2)

Figure 5.2: Time Accuracy plots in the configuration (□), velocity (△), and acceleration (○) within the PCE-ET GSSSS family of algorithms with $\eta_3 = 1$ for the linear dissipative nonhomogeneous system, $\ddot{q} + \dot{q} + 10q = 0.5\sin(t)$, with the initial conditions $q_0 = 1$ and $\dot{q}_0 = 1$ - Single-field form



(a) U0V0/V0U0(1.0,1.0,1.0)

(b) U0V0(1.0,1.0,0.0)

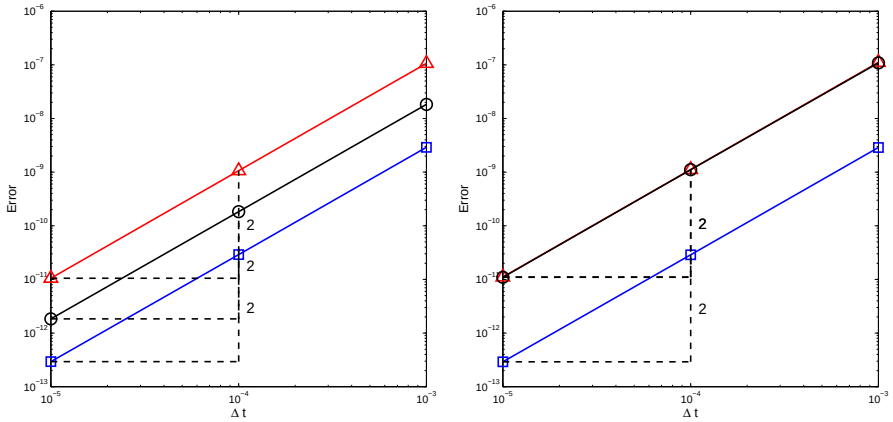
(c) V0U0(1.0,1.0,0.0)

(d) U0V0/V0U0(0.8,1.0,0.8)

(e) U0V1(0.6,0.9,0.2)

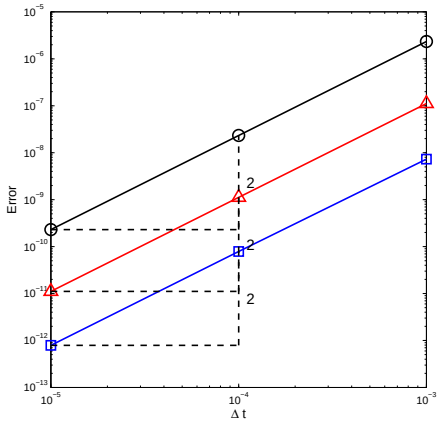
(f) V0U1(0.6,0.9,0.2)

Figure 5.3: Time Accuracy plots in the configuration (□), velocity (△), and acceleration (○) within the PCE-IT GSSSS family of algorithms with $\eta_3 = 0$ for the linear dissipative nonhomogeneous system, $\ddot{q} + \dot{q} + 10q = 0.5\sin(t)$, with the initial conditions $q_0 = 1$ and $\dot{q}_0 = 1$ - Single-field form

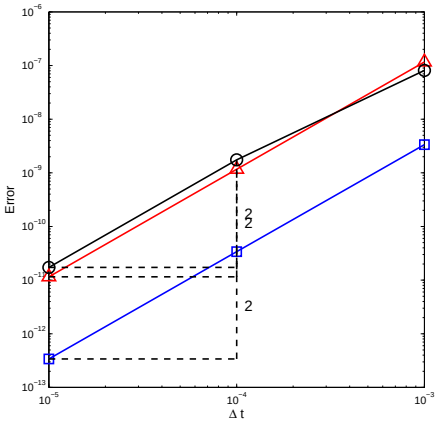


(a) U0V0/V0U0(1.0,1.0,1.0)

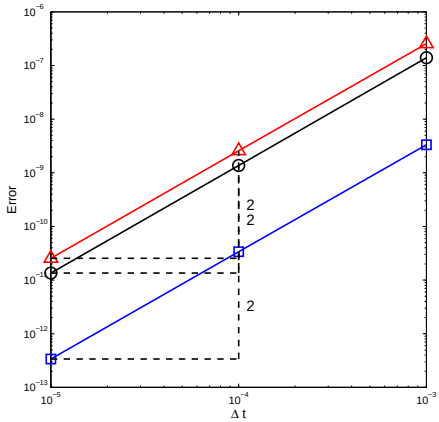
(b) U0V0(1.0,1.0,0.0)



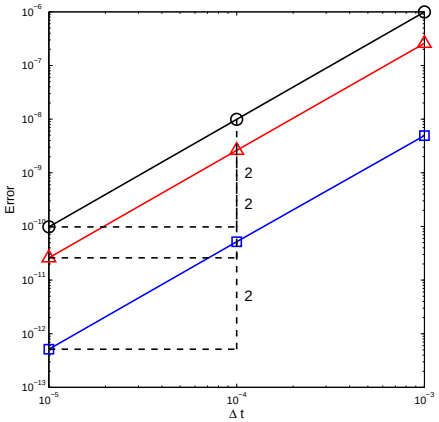
(c) V0U0(1.0,1.0,0.0)



(d) U0V0/V0U0(0.8,1.0,0.8)

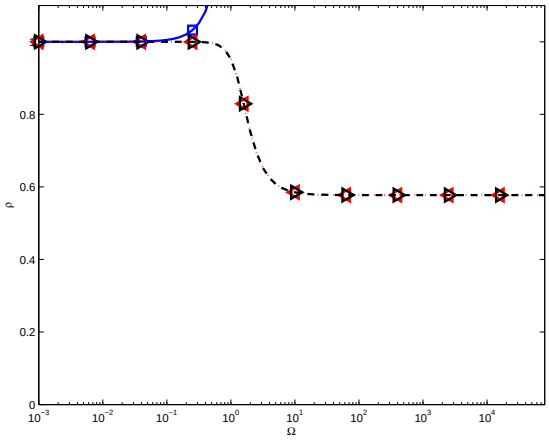


(e) U0V1(0.6,0.9,0.2)

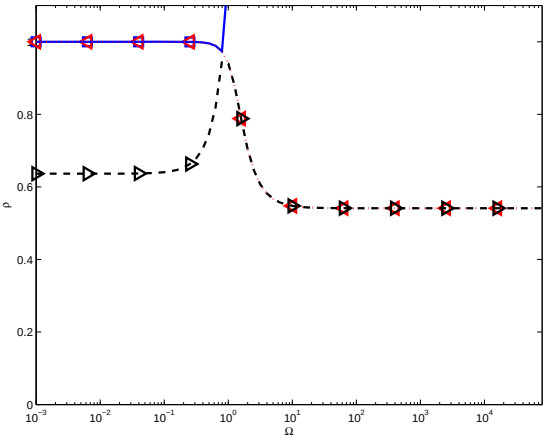


(f) V0U1(0.6,0.9,0.2)

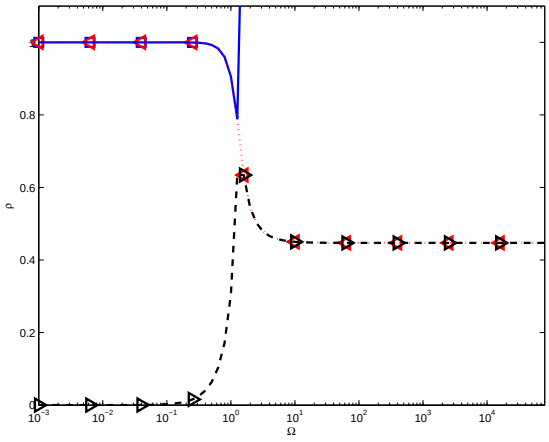
Figure 5.4: Time Accuracy plots in the configuration (□), velocity (△), and acceleration (○) within the PCE-ET GSSSS family of algorithms with $\eta_3 = 0$ for the linear dissipative nonhomogeneous system, $\ddot{q} + \dot{q} + 10q = 0.5\sin(t)$, with the initial conditions $q_0 = 1$ and $\dot{q}_0 = 1$ - Single-field form



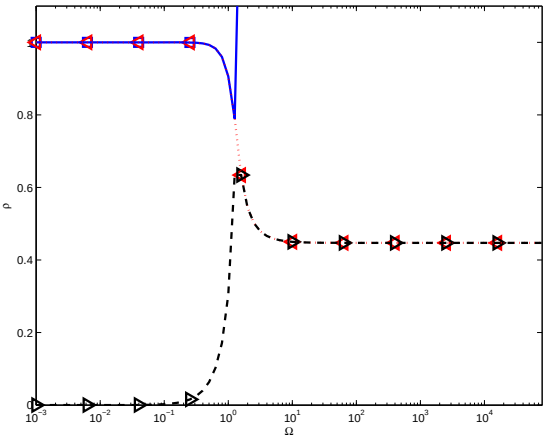
(a) U0V0/V0U0(1.0,1.0,1.0)



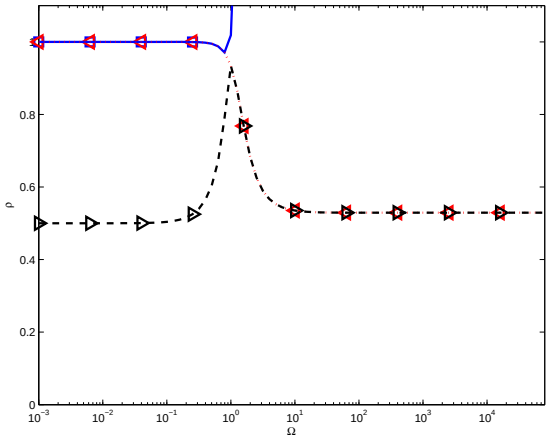
(b) U0V0/V0U0(0.8,1.0,0.8)



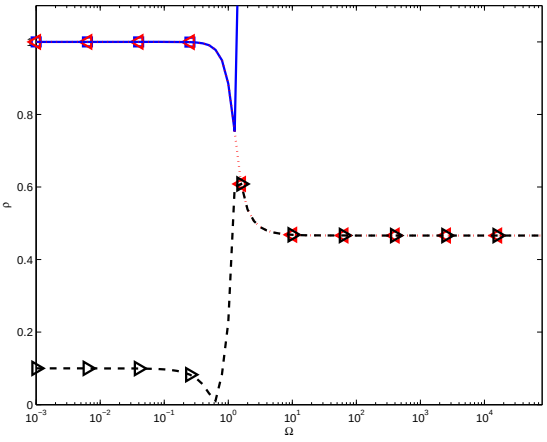
(c) U0V0(1.0,1.0,0.0)



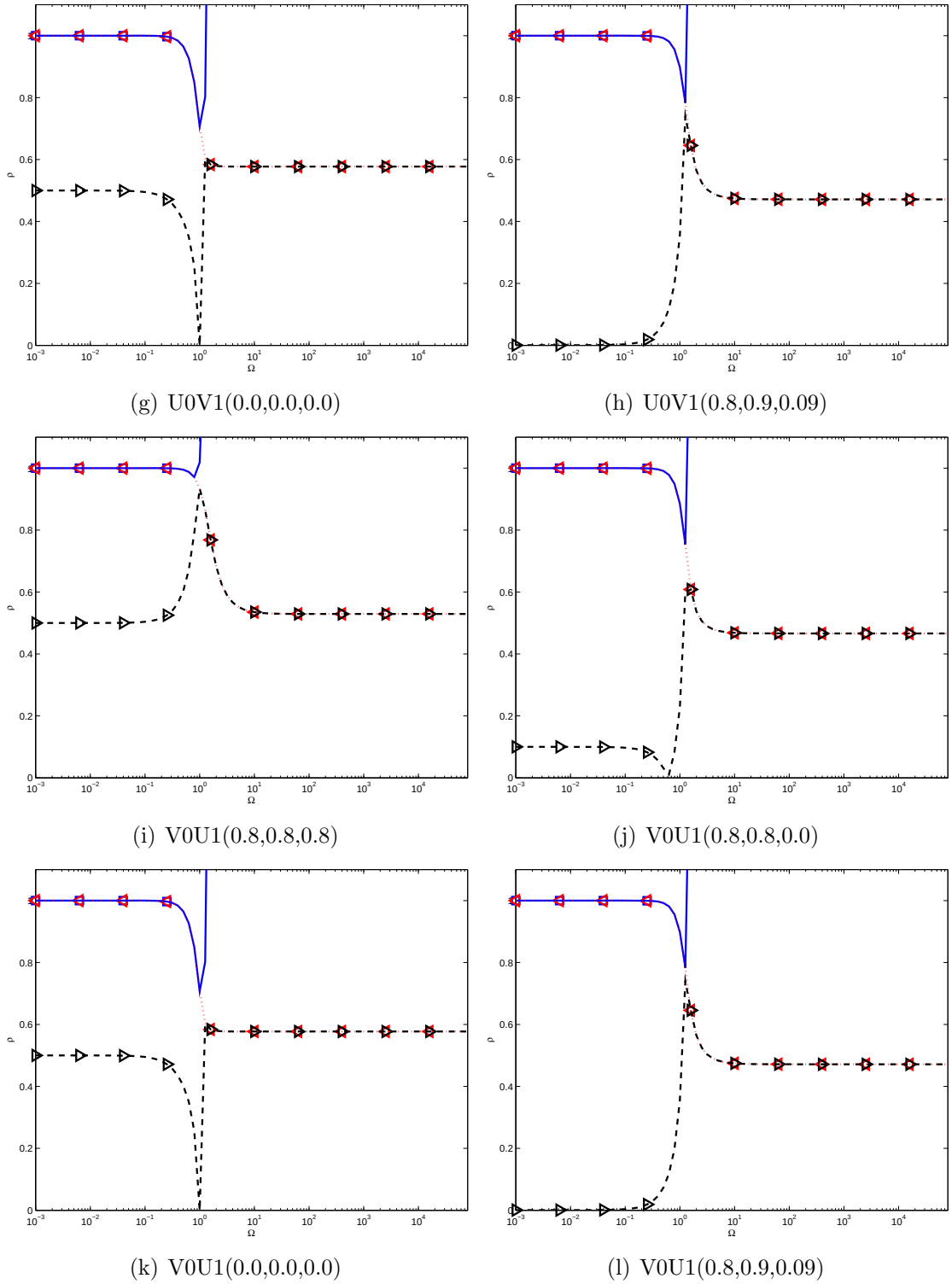
(d) V0U0(1.0,1.0,0.0)



(e) U0V1(0.8,0.8,0.8)



(f) U0V1(0.8,0.8,0.0)

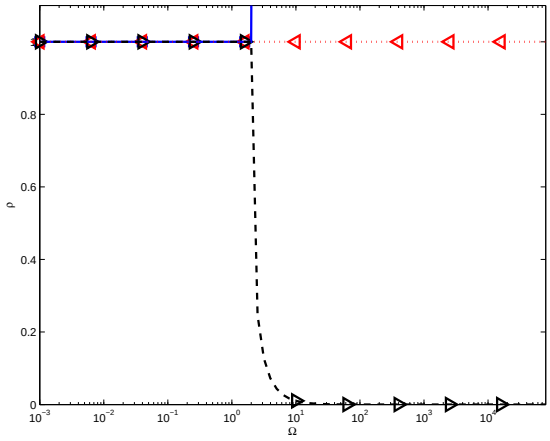


(g) U0V1(0.0,0.0,0.0) (h) U0V1(0.8,0.9,0.09)

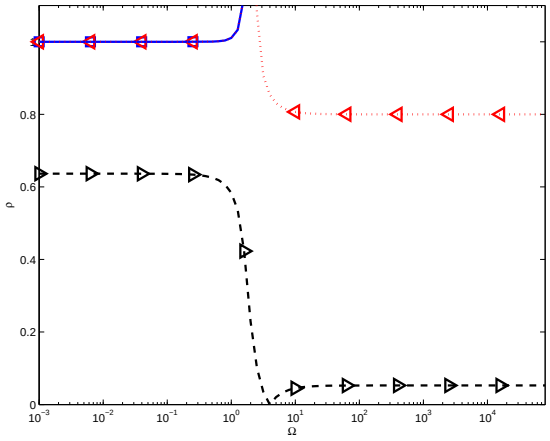
(i) V0U1(0.8,0.8,0.8) (j) V0U1(0.8,0.8,0.0)

(k) V0U1(0.0,0.0,0.0) (l) V0U1(0.8,0.9,0.09)

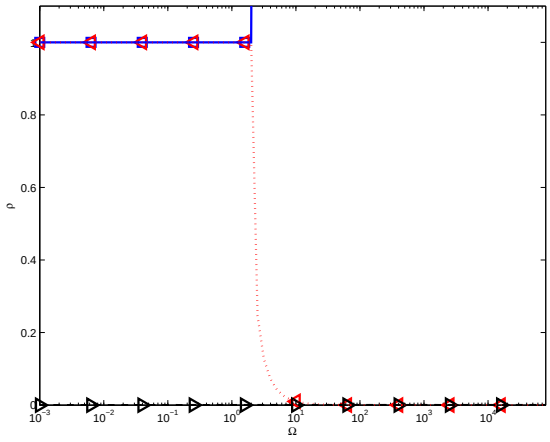
Figure 5.5: Stability plots of selected algorithms within Algorithms 8 and 9 with $\eta_3 = 1$ in the conservative system, $\ddot{q} + q = 0$ ($\omega = 1$) - Single-field form



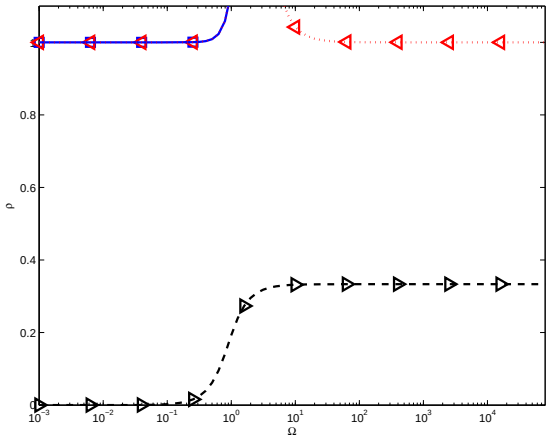
(a) U0V0/V0U0(1.0,1.0,1.0)



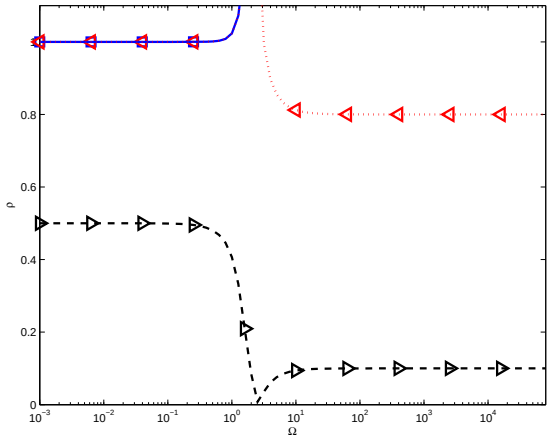
(b) U0V0/V0U0(0.8,1.0,0.8)



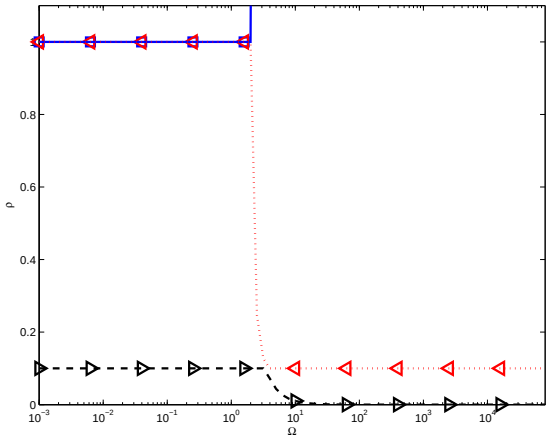
(c) U0V0(1.0,1.0,0.0)



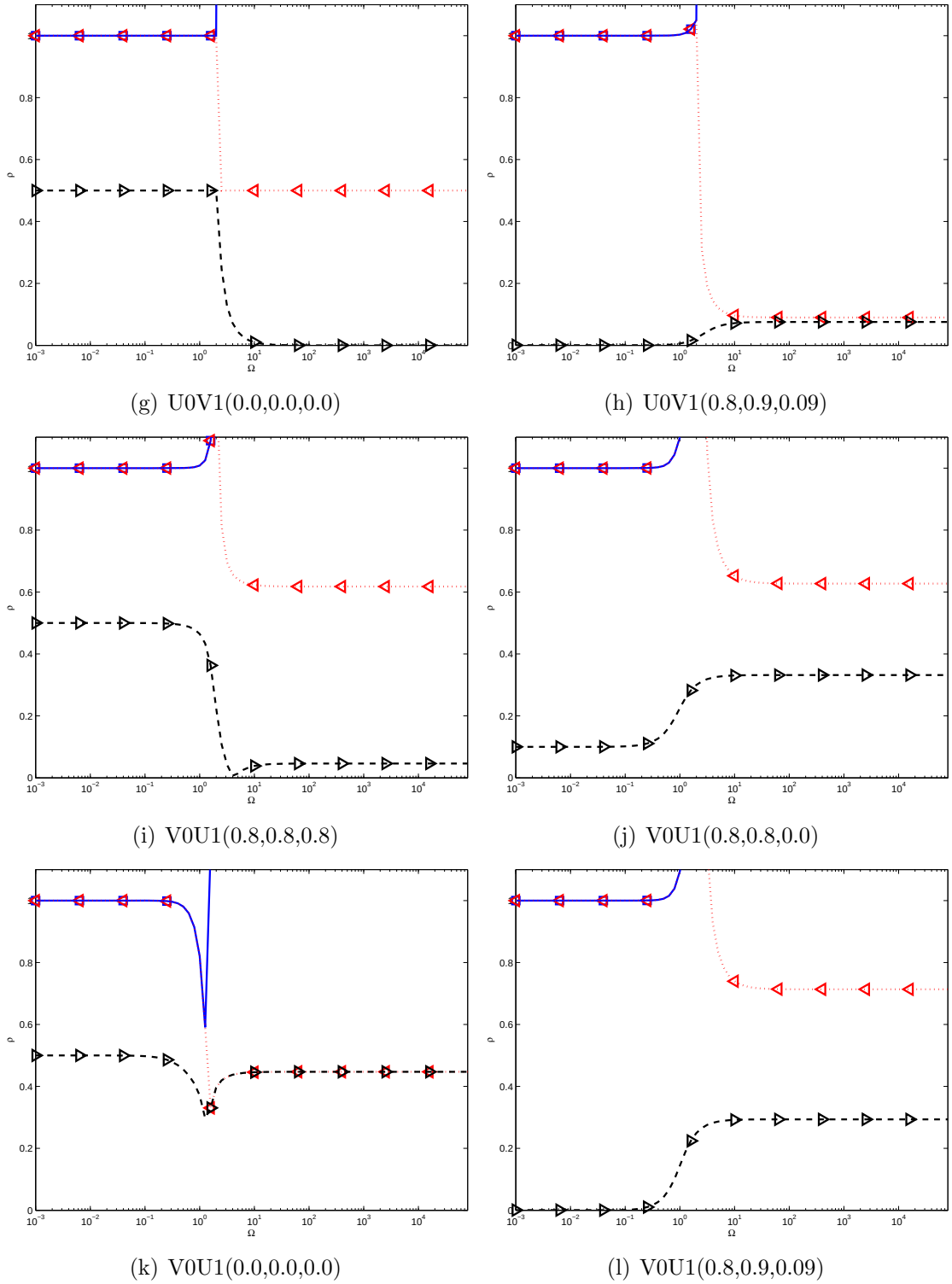
(d) V0U0(1.0,1.0,0.0)



(e) U0V1(0.8,0.8,0.8)



(f) U0V1(0.8,0.8,0.0)

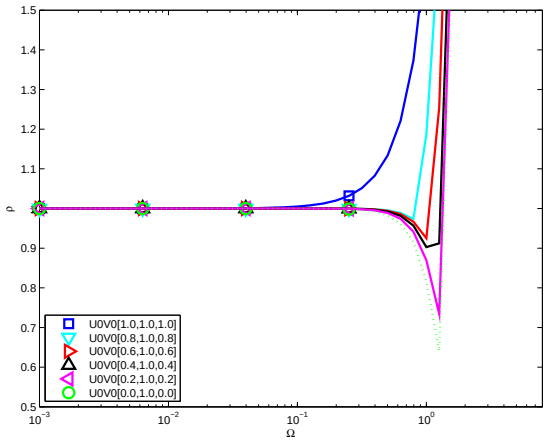


(g) U0V1(0.0,0.0,0.0) (h) U0V1(0.8,0.9,0.09)

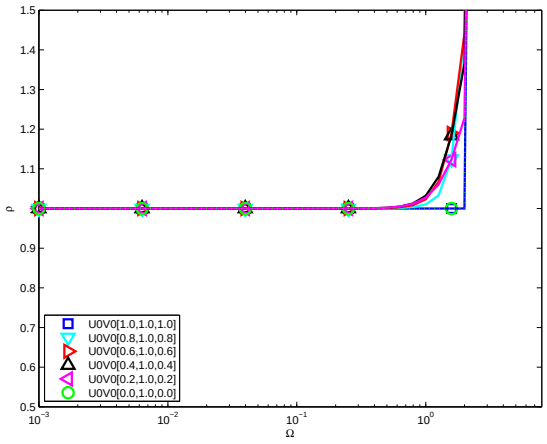
(i) V0U1(0.8,0.8,0.8) (j) V0U1(0.8,0.8,0.0)

(k) V0U1(0.0,0.0,0.0) (l) V0U1(0.8,0.9,0.09)

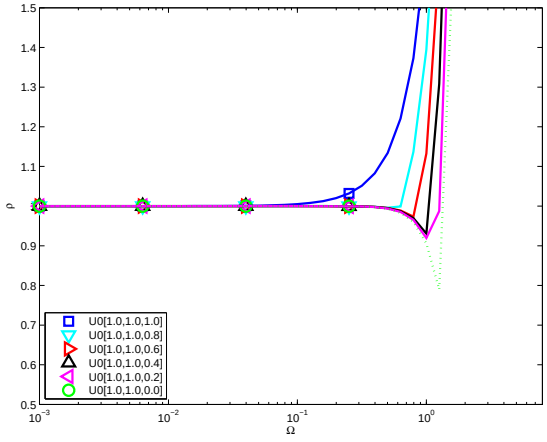
Figure 5.6: Stability plots of selected algorithms within Algorithms 8 and 9 with $\eta_3 = 0$ in the conservative system, $\ddot{q} + q = 0$ ($\omega = 1$) - Single-field form



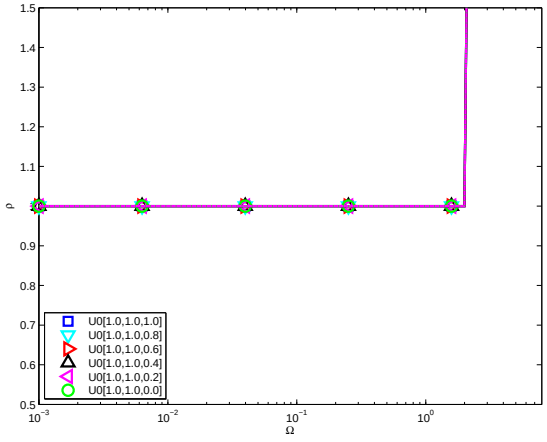
(a) U0V0/V0U0 Optimal: $\eta_3 = 1$



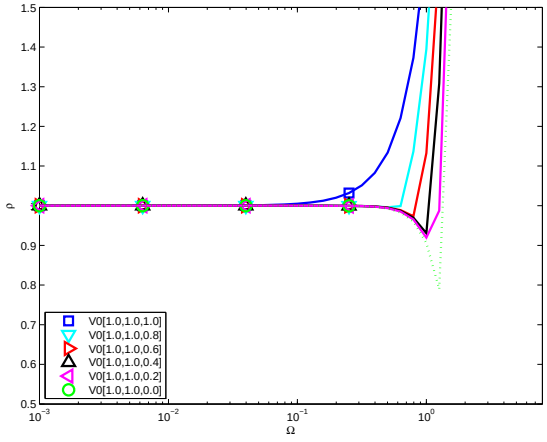
(b) U0V0/V0U0 Optimal: $\eta_3 = 0$



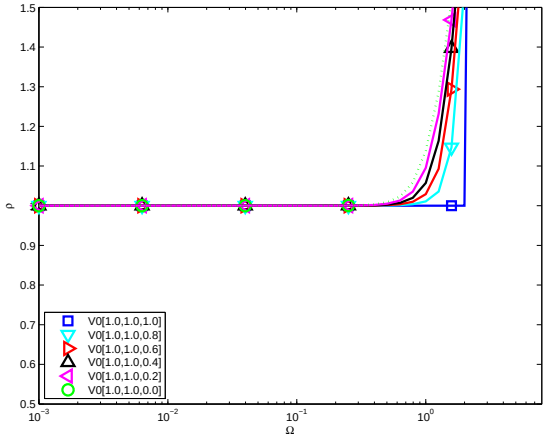
(c) U0V0(1,1,ρ^s_∞): $\eta_3 = 1$



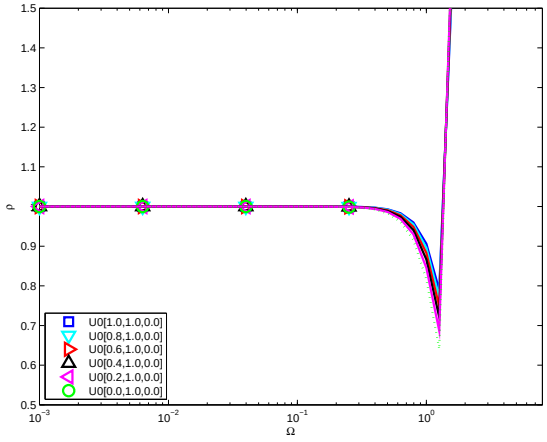
(d) U0V0(1,1,ρ^s_∞): $\eta_3 = 0$



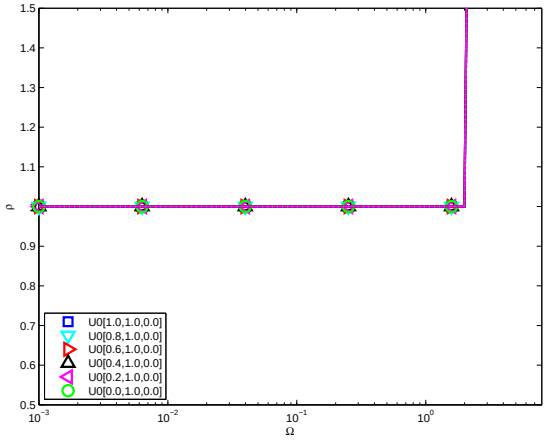
(e) V0U0(1,1,ρ^s_∞): $\eta_3 = 1$



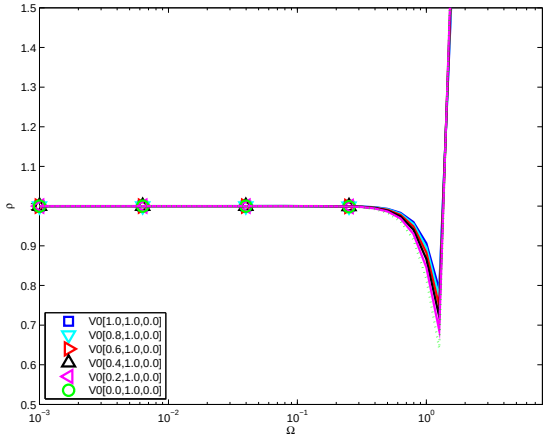
(f) V0U0(1,1,ρ^s_∞): $\eta_3 = 0$



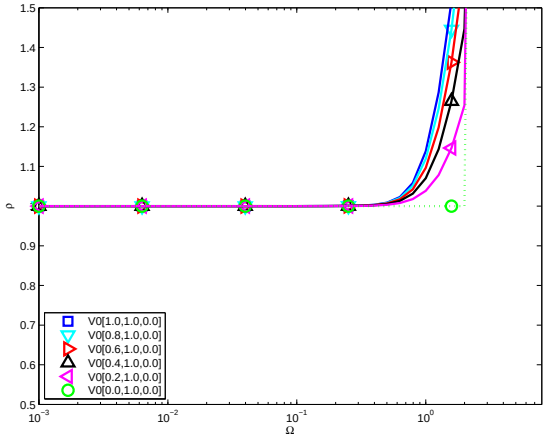
(g) U0V0($\rho_\infty^{\min}$,1,0): $\eta_3 = 1$



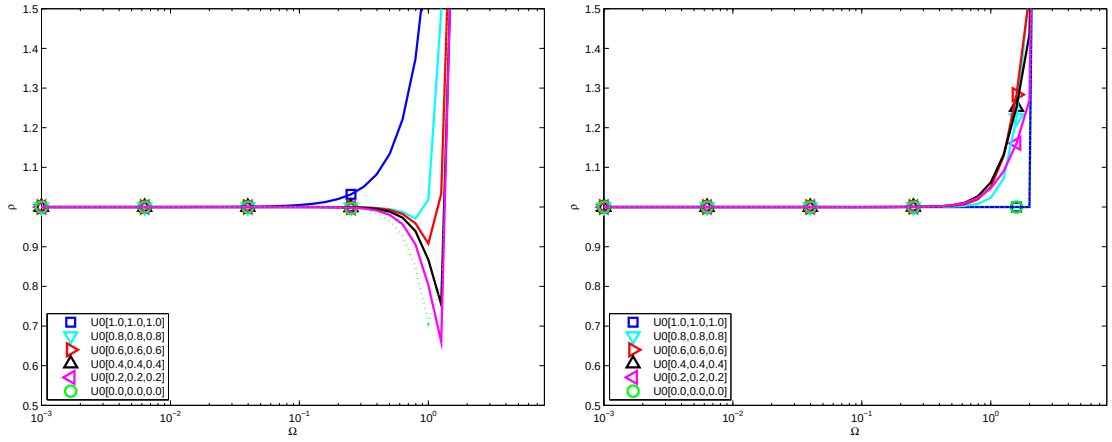
(h) U0V0($\rho_\infty^{\min}$,1,0): $\eta_3 = 0$



(i) V0U0($\rho_\infty^{\min}$,1,0): $\eta_3 = 1$

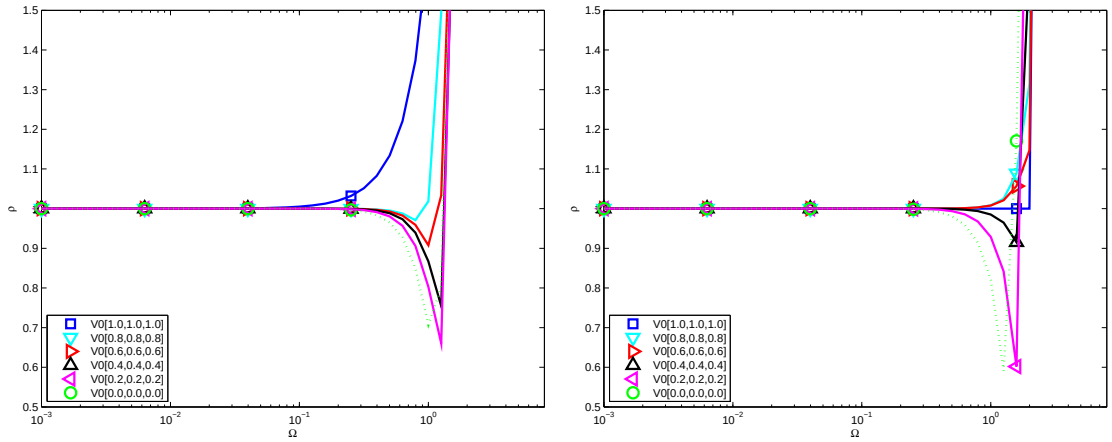


(j) V0U0($\rho_\infty^{\min}$,1,0): $\eta_3 = 0$



(k) U0V1(ρ_∞,ρ_∞,ρ_∞): $\eta_3 = 1$

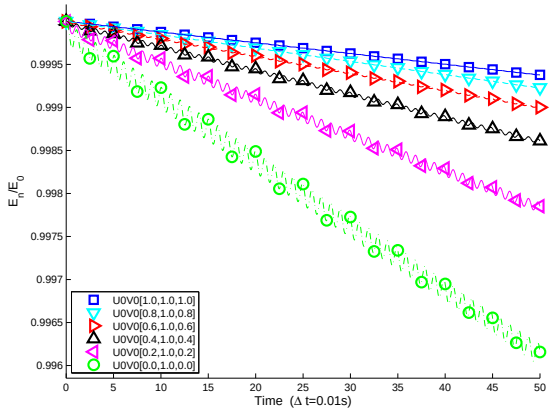
(l) U0V1(ρ_∞,ρ_∞,ρ_∞): $\eta_3 = 0$



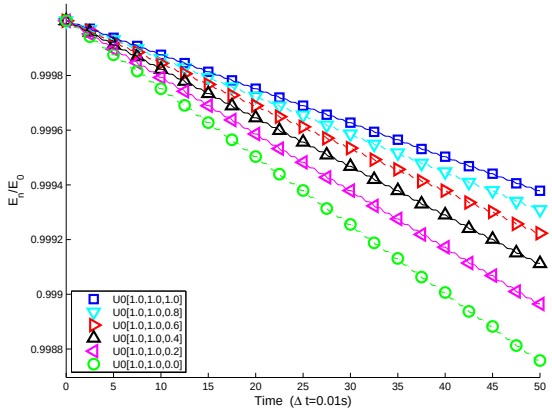
(m) V0U1(ρ_∞,ρ_∞,ρ_∞): $\eta_3 = 1$

(n) V0U1(ρ_∞,ρ_∞,ρ_∞): $\eta_3 = 0$

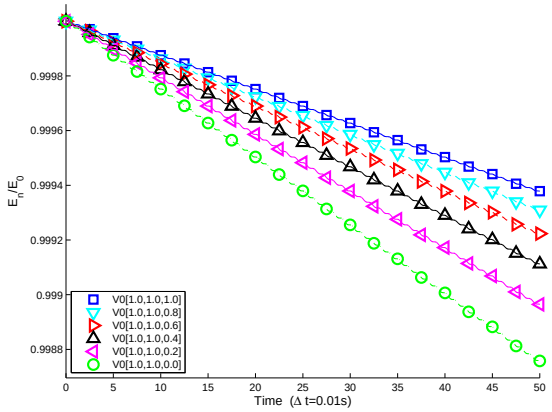
Figure 5.7: Spectral radius ($\rho_\infty^{\max}$) plots of selected algorithms within the explicit family of GSSSS algorithms for $\ddot{q} + q = 0$ ($\omega = 1$) - Single-field form



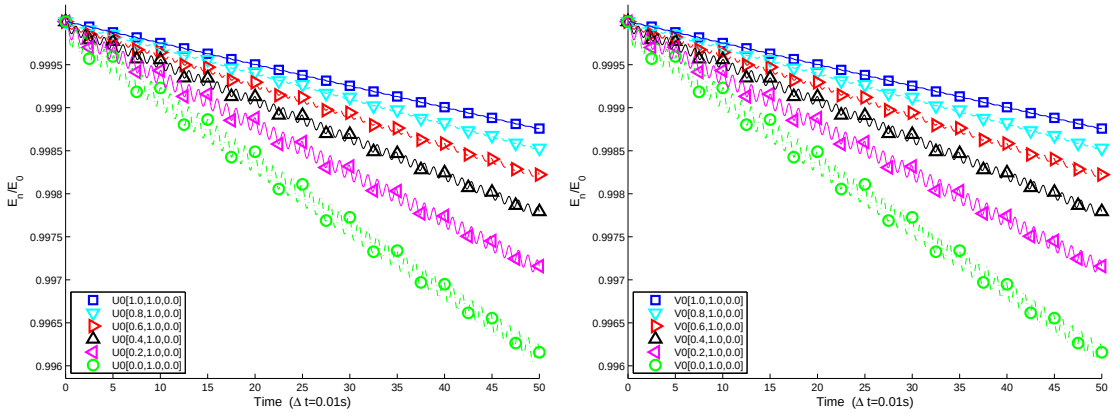
(a) UU0V0/V0U0 Optimal



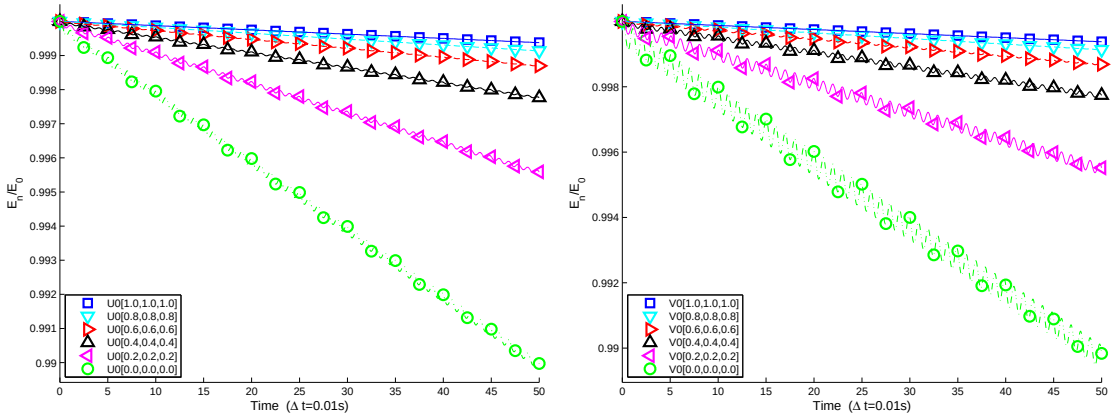
(b) U0V0(1,1,ρ^{s}_{∞})



(c) V0U0(1,1,ρ^{s}_{∞})

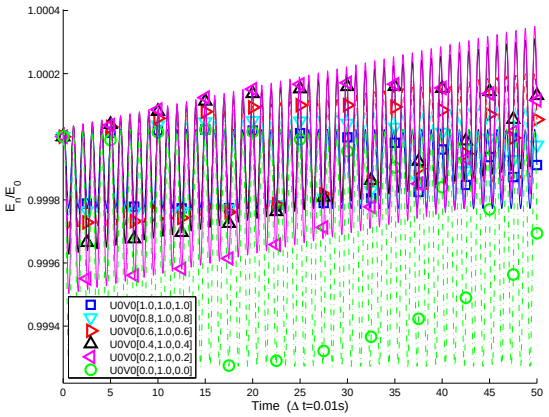


(d) U0V0($\rho_\infty^{\min}$,1,0) (e) V0U0($\rho_\infty^{\min}$,1,0)

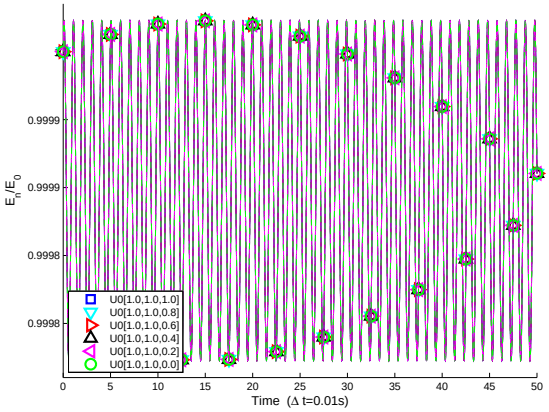


(f) U0V1(ρ_∞,ρ_∞,ρ_∞) (g) V0U1(ρ_∞,ρ_∞,ρ_∞)

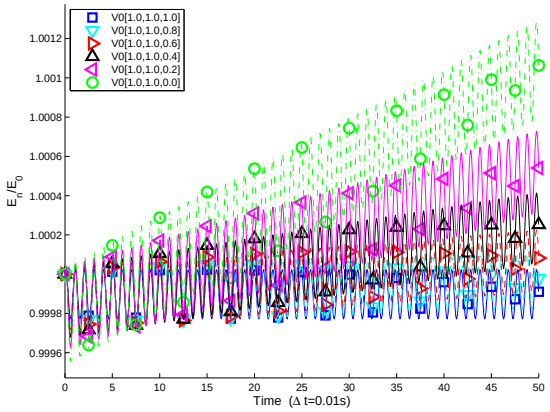
Figure 5.8: Mechanical Energy plots of the selected algorithms within Algorithms 8 and 9 with $\eta_3 = 1$ ($\Delta t = 0.01$ sec) for the linear conservative homogeneous system, $\ddot{q} + 10q = 0$, with the initial conditions $q_0 = 1$ and $\dot{q}_0 = 1$ - Single-field form



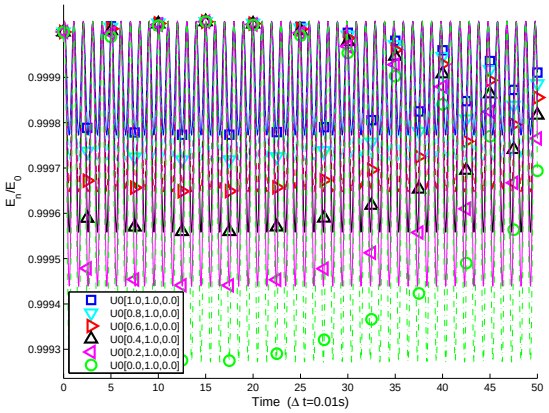
(a) U0V0/V0U0 Optimal



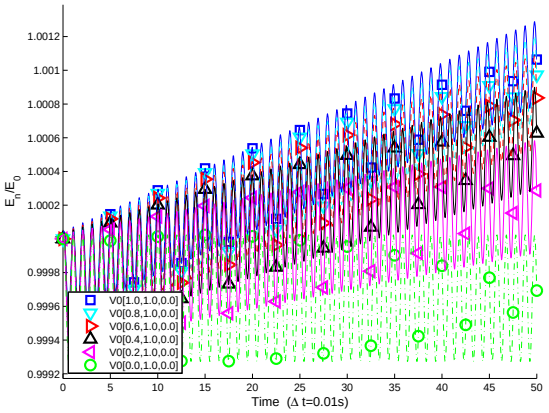
(b) U0V0(1,1,ρ^{s}_{∞})



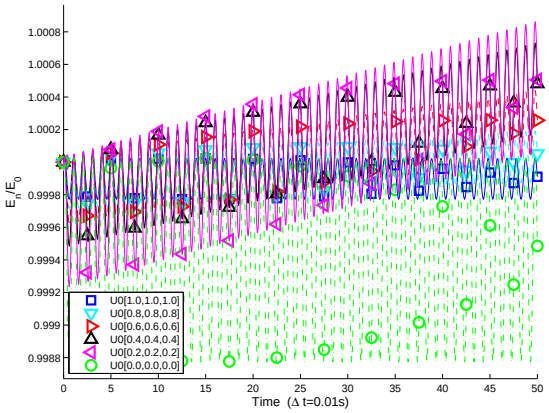
(c) V0U0(1,1,ρ^{s}_{∞})



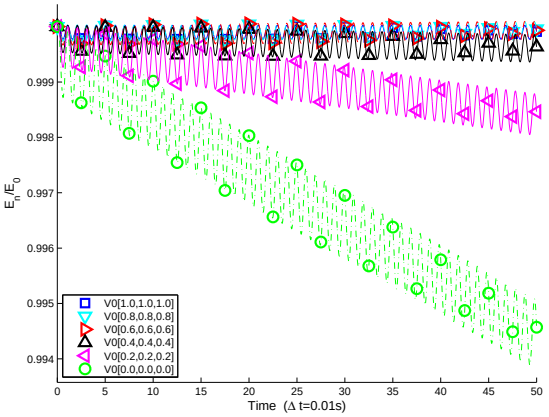
(d) U0V0($\rho_\infty^{\min}$,1,0)



(e) V0U0($\rho_\infty^{\min}$,1,0)

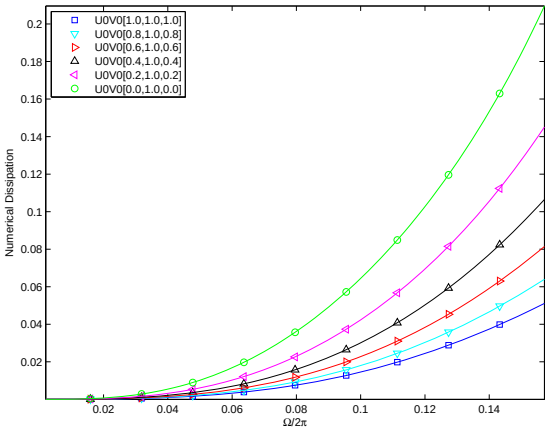


(f) U0V1(ρ_∞,ρ_∞,ρ_∞)

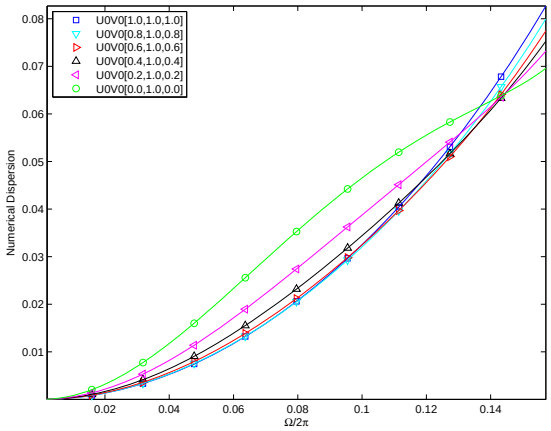


(g) V0U1(ρ_∞,ρ_∞,ρ_∞)

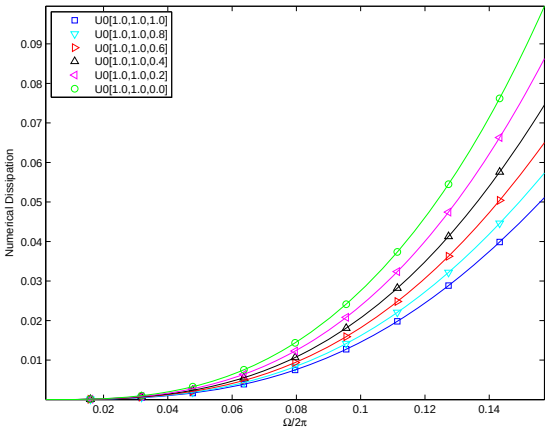
Figure 5.9: Mechanical Energy plots of the selected algorithms within Algorithms 8 and 9 with $\eta_3 = 0$ ($\Delta t = 0.01$ sec) for the linear conservative homogeneous system, $\ddot{q} + 10q = 0$, with the initial conditions $q_0 = 1$ and $\dot{q}_0 = 1$ - Single-field form



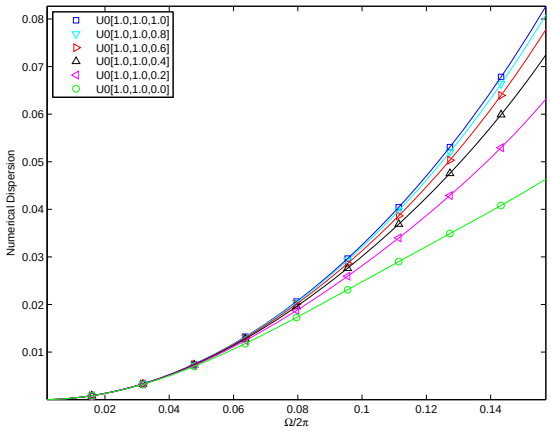
(a) U0V0/V0U0 Optimal: Dissipation



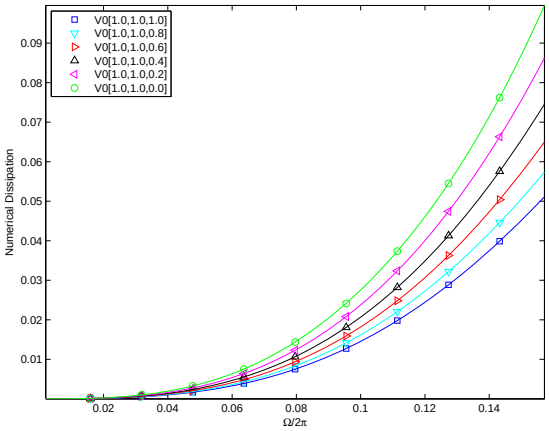
(b) U0V0/V0U0 Optimal: Dispersion



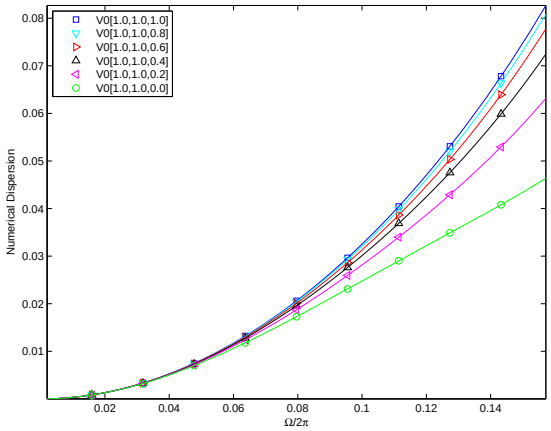
(c) U0V0(1,1,ρ^{s}_{∞}): Dissipation



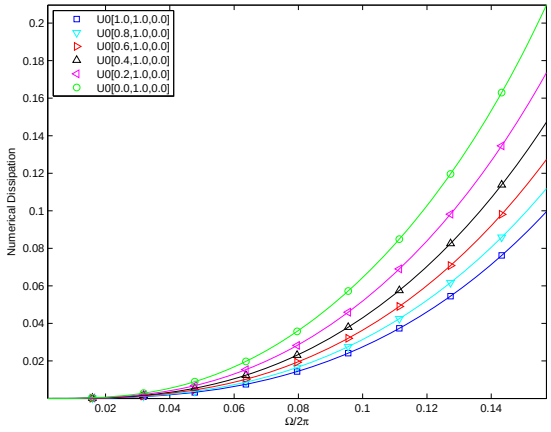
(d) U0V0(1,1,ρ^{s}_{∞}): Dispersion



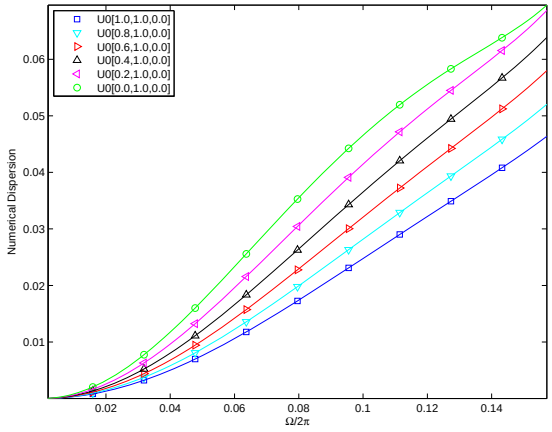
(e) V0U0(1,1,ρ^{s}_{∞}): Dissipation



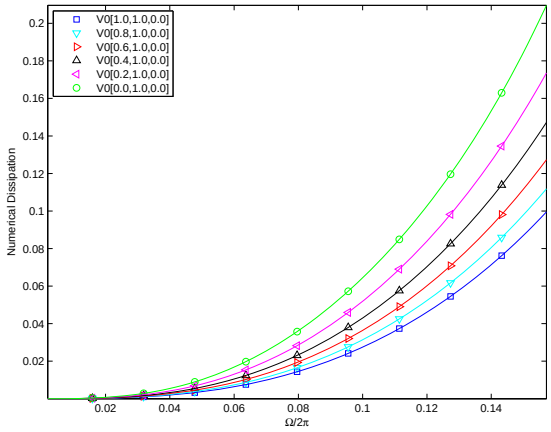
(f) V0U0(1,1,ρ^{s}_{∞}): Dispersion



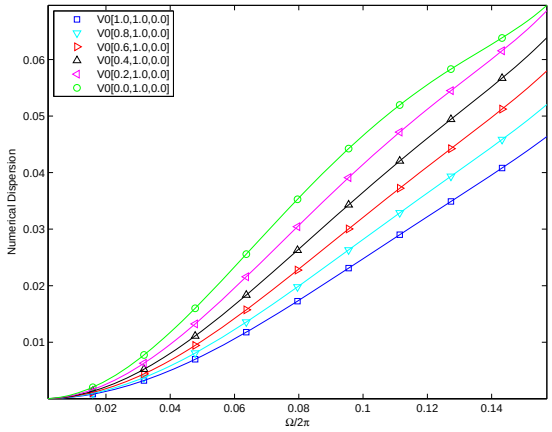
(g) U0V0($\rho_\infty^{\min}$,1,0): Dissipation



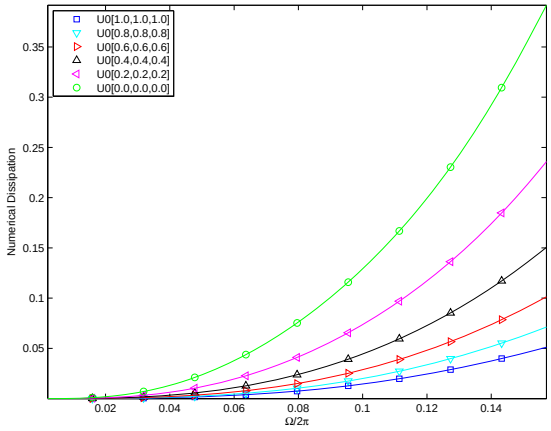
(h) U0V0($\rho_\infty^{\min}$,1,0): Dispersion



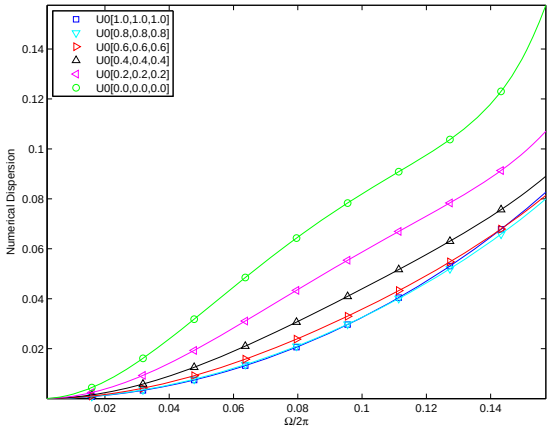
(i) V0U0($\rho_\infty^{\min}$,1,0): Dissipation



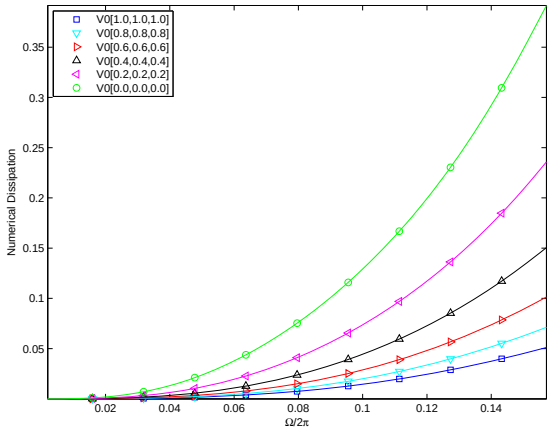
(j) V0U0($\rho_\infty^{\min}$,1,0): Dispersion



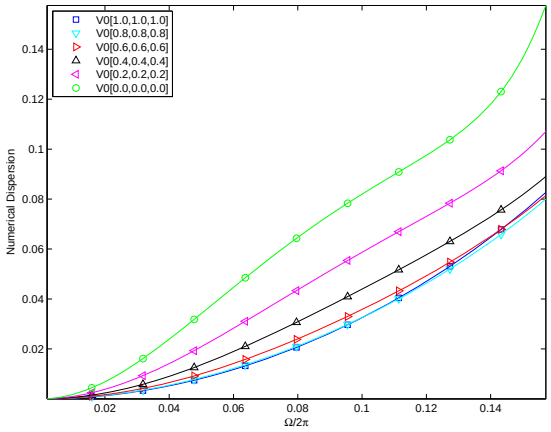
(k) U0V1(ρ_∞,ρ_∞,ρ_∞): Dissipation



(l) U0V1(ρ_∞,ρ_∞,ρ_∞): Dispersion

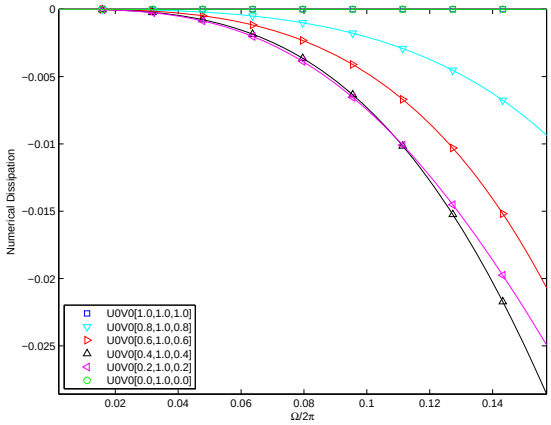


(m) V0U1(ρ_∞,ρ_∞,ρ_∞): Dissipation

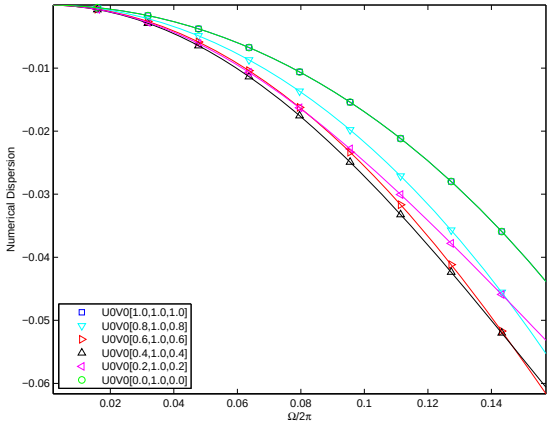


(n) V0U1(ρ_∞,ρ_∞,ρ_∞): Dispersion

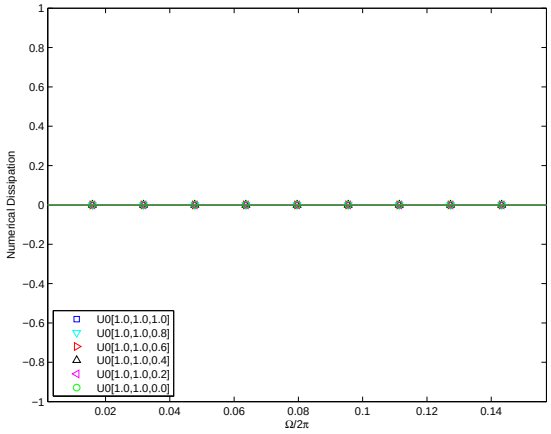
Figure 5.10: Numerical dissipation and numerical dispersion plots of selected algorithms within Algorithms 8 and 9 with $\eta_3 = 1$ in the conservative system, $\ddot{q}+q = 0$ ($\omega = 1$) - Single-field form



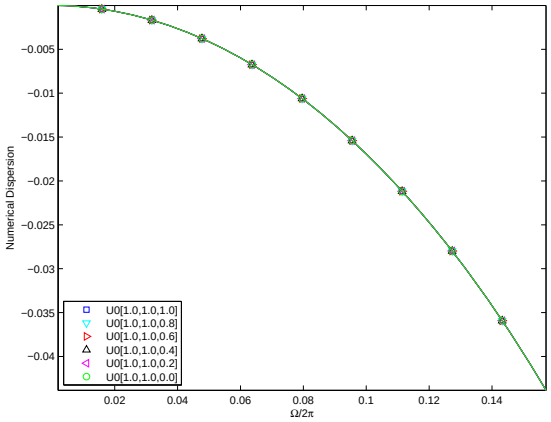
(a) U0V0/V0U0 Optimal: Dissipation



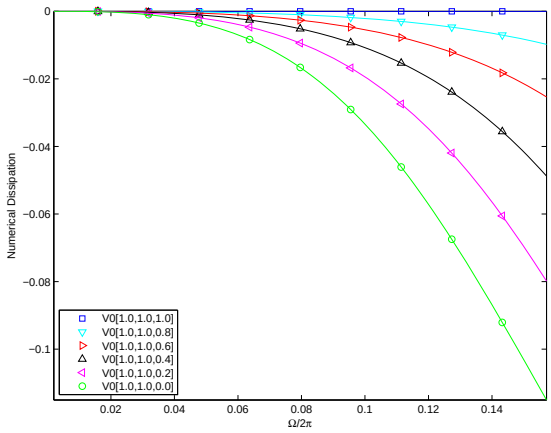
(b) U0V0/V0U0 Optimal: Dispersion



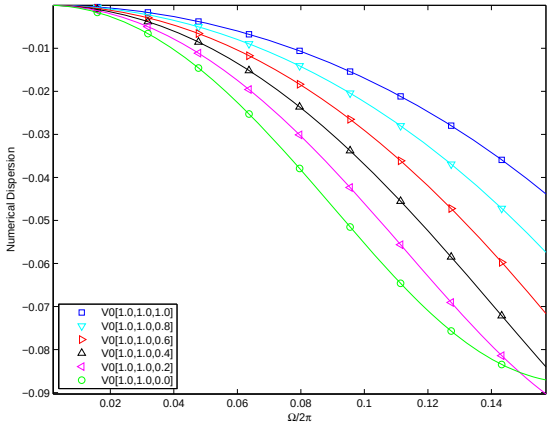
(c) U0V0(1,1,ρ^{s}_{∞}): Dissipation



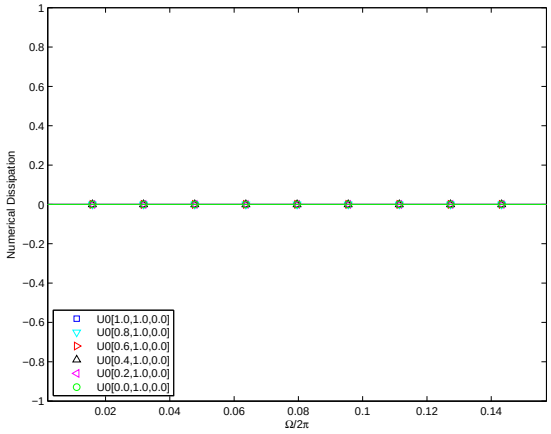
(d) U0V0(1,1,ρ^{s}_{∞}): Dispersion



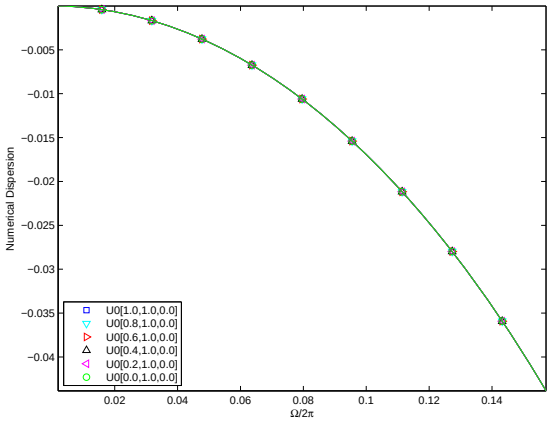
(e) V0U0(1,1,ρ^{s}_{∞}): Dissipation



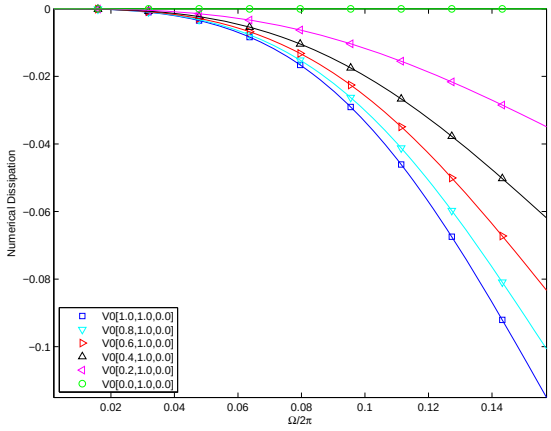
(f) V0U0(1,1,ρ^{s}_{∞}): Dispersion



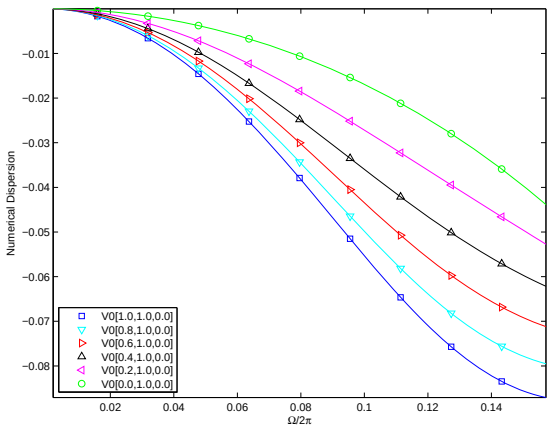
(g) U0V0($\rho_\infty^{\min}$,1,0): Dissipation



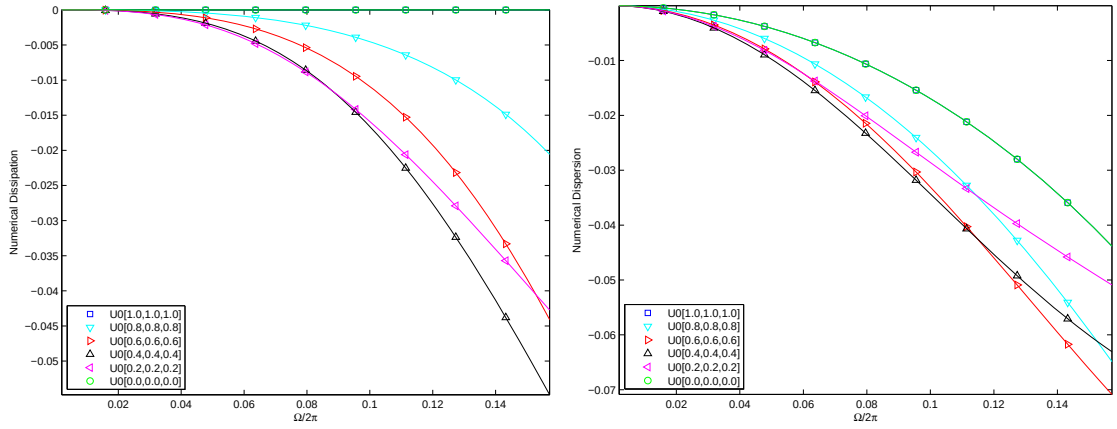
(h) U0V0($\rho_\infty^{\min}$,1,0): Dispersion



(i) V0U0($\rho_\infty^{\min}$,1,0): Dissipation



(j) V0U0($\rho_\infty^{\min}$,1,0): Dispersion



(k) U0V1(ρ_∞,ρ_∞,ρ_∞): Dissipation

(l) U0V1(ρ_∞,ρ_∞,ρ_∞): Dispersion

(m) V0U1(ρ_∞,ρ_∞,ρ_∞): Dissipation

(n) V0U1(ρ_∞,ρ_∞,ρ_∞): Dispersion

Figure 5.11: Numerical dissipation and numerical dispersion plots of selected algorithms within Algorithms 8 and 9 with $\eta_3 = 0$ in the conservative system, $\ddot{q}+q = 0$ ($\omega = 1$) - Single-field form

Chapter 6

Implicit Generalized Single Step Single Solve (I-GSSSS) Algorithms in Two- and Single-Field Forms for Nonlinear Dynamical Systems (N-body Systems and Elastodynamics)

In this chapter, we show the direct extensions of the linear dynamical algorithms which are a necessary first step to nonlinear dynamical systems. This is accomplished via two schools of thought, namely, the ***classical time weighted residual methodology*** and the new ***normalized time weighted residual methodology***, taking the parent algorithms in linear dynamical systems as the underlying bases in the two- and single-field forms . It is to be noted that the former (classical) has limitations for nonlinear dynamical systems and is not the proper way; while the latter (new normalized) circumvents the deficiencies associated with the classical approach, and is the proper way to provide extensions

to nonlinear dynamical systems. How to provide proper extensions to nonlinear dynamical systems from the development of the parent algorithms designed in linear dynamical systems via the time weighted residual methodology is not at all trivial. We provide a sound theoretical basis to show the fundamental developments. In particular, the algorithmic time level, discussed previously in the linear dynamical system, plays an important role. In the following, we show the extension to nonlinear dynamical systems via the classical and normalized time weighted residual methodology, taking Algorithm 1 as the parent algorithm.

6.1 Nonlinear I-GSSSS Algorithms and Framework in Two-Field Form

The standard equation of motion in the two-field form in nonlinear dynamical systems takes the form of the following initial-value problem,

$$\boxed{\begin{aligned}&\textbf{\textit{Balance Equation:}}\\&\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{f}^{\text{int}}(\mathbf{q},\boldsymbol{\nu}) = \mathbf{f}^{\text{ext}}(\mathbf{q},\boldsymbol{\nu},t)\\&\textbf{\textit{Kinematic constraint:}}\\&\dot{\mathbf{q}}(t) = \boldsymbol{\nu}(t) \quad \forall t \in \mathbb{I}\\&\textbf{\textit{Initial conditions:}}\\&\mathbf{q}(t_0) = \mathbf{q}_0\\&\boldsymbol{\nu}(t_0) = \boldsymbol{\nu}_0\end{aligned}} \tag{6.1}$$

where $\mathbf{f}^{\text{int}}(\mathbf{q},\boldsymbol{\nu}) : TQ \to \mathbb{R}^{3N}$ and $\mathbf{f}^{\text{ext}}(\mathbf{q},\boldsymbol{\nu},t) : TQ \times \mathbb{I} \to \mathbb{R}^{3N}$ are the internal and external forces, respectively. Suppose the internal force is given only with respect to the configuration $\mathbf{q}$, i.e., $\mathbf{f}^{\text{int}}(\mathbf{q}) : Q \to \mathbb{R}^{3N}$, and the external force is given as the summation of the time-dependent external force, $\mathbf{f}(t) : \mathbb{I} \to \mathbb{R}^{3N}$, and the external force vector which does not depend on time explicitly, $\mathbf{f}^{\text{e}}(\mathbf{q},\boldsymbol{\nu}) : TQ \to \mathbb{R}^{3N}$. If $\mathbf{f}^{\text{e}}(\mathbf{q},\boldsymbol{\nu})$ consists of the dissipative force, $\mathbf{f}^{\text{diss}}(\mathbf{q},\boldsymbol{\nu}) : TQ \to \mathbb{R}^{3N}$, and the external

conservative force, $\mathbf{f}^{\mathrm{c}}(\mathbf{q}) : Q \to \mathbb{R}^{3N}$, the balance equation in Eq. (6.1) can be written as

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{f}^{\mathrm{int}}(\mathbf{q}) = \mathbf{f}(t) + \mathbf{f}^{\mathrm{diss}}(\mathbf{q}, \boldsymbol{\nu}) + \mathbf{f}^{\mathrm{c}}(\mathbf{q}) \tag{6.2}$$

Note that the internal (conservative) force is defined from the internal potential energy function, $\mathscr{U}^{\mathrm{int}}(\mathbf{q}) : Q \to \mathbb{R}$, as $\mathbf{f}^{\mathrm{int}}(\mathbf{q}) := \nabla \mathscr{U}^{\mathrm{int}}(\mathbf{q})$; while the external conservative force is defined from the external potential energy functions, $\mathscr{U}^{\mathrm{ext}}(\mathbf{q}) : Q \to \mathbb{R}$, as $\mathbf{f}^{\mathrm{c}}(\mathbf{q}) := -\nabla \mathscr{U}^{\mathrm{ext}}(\mathbf{q})$. Similarly, the dissipative force is defined from the dissipative potential $\vartheta(\mathbf{q}, \dot{\mathbf{q}}) : TQ \to \mathbb{R}$ as $\mathbf{f}^{\mathrm{diss}}(\mathbf{q}, \boldsymbol{\nu}) := -\partial\vartheta/\partial\dot{\mathbf{q}}$. Defining the force vector $\mathbf{N}(\mathbf{q}, \boldsymbol{\nu}) : TQ \to \mathbb{R}^{3N}$ as $\mathbf{N}(\mathbf{q}, \boldsymbol{\nu}) := \mathbf{f}^{\mathrm{int}}(\mathbf{q}) - \mathbf{f}^{\mathrm{c}}(\mathbf{q}) - \mathbf{f}^{\mathrm{diss}}(\mathbf{q}, \boldsymbol{\nu})$, Eq. (6.2) yields

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{N}(\mathbf{q}, \boldsymbol{\nu}) = \mathbf{f}(t) \tag{6.3}$$

If the dissipative force depends only upon $\boldsymbol{\nu}$ and given linearly as $\mathbf{f}^{\mathrm{diss}}(\boldsymbol{\nu}) = -\mathbf{C}\boldsymbol{\nu}$, Eq. (6.3) can be written as

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}\boldsymbol{\nu} + \mathbf{f}^{\mathrm{pot}}(\mathbf{q}) = \mathbf{f}(t) \tag{6.4}$$

where $\mathbf{f}^{\mathrm{pot}}(\mathbf{q}) := \mathbf{f}^{\mathrm{int}}(\mathbf{q}) - \mathbf{f}^{\mathrm{c}}(\mathbf{q}) = \nabla \mathscr{U}$ is the potential force.

6.1.1 Option I: Algorithms and Designs Via Classical Time Weighted Residual Methodology

Assuming $\mathbf{N}(t) : \mathbb{I} \to \mathbb{R}^{3N}$, consider the nonlinear two-field form dynamical system,

Balance Equation:
$\mathbf{M}\dot{\boldsymbol{\nu}}(t) + \mathbf{N}(t) = \mathbf{f}(t)$
Kinematic constraint:
$\dot{\mathbf{q}}(t) = \boldsymbol{\nu}(t) \quad \forall t \in \mathbb{I}$
Initial conditions:
$\mathbf{q}(t_0) = \mathbf{q}_0$
$\boldsymbol{\nu}(t_0) = \boldsymbol{\nu}_0$

(6.5)

From the balance equation in Eq. (6.5), we have

$$\mathbf{M}\dot{\boldsymbol{\nu}}(t) + \mathbf{N}(t) - \mathbf{f}(t) = \mathbf{0} \;\; \forall t \in \mathbb{I} \tag{6.6}$$

The asymptotic series expansions of $\mathbf{N}$, $\boldsymbol{\nu}$, and $\dot{\boldsymbol{\nu}}$, about time $t = t_{n+\alpha} := (1-\alpha)t_n + \alpha t_{n+1} \in [t_n, t_{n+1}]$ for $\alpha \in [0,1]$, yields

$$\begin{aligned} \mathbf{N}(t_{n+\alpha}) &\cong \mathbf{N}(t_n) + \Theta_1 \dot{\mathbf{N}}(t_n)[t_{n+\alpha} - t_n] \\ &\quad + \Theta_2 \frac{\dot{\mathbf{N}}(t_{n+1}) - \dot{\mathbf{N}}(t_n)}{\Delta t}[t_{n+\alpha} - t_n]^2 \qquad (6.7) \\ \dot{\boldsymbol{\nu}}(t_{n+\alpha}) &\cong \frac{\boldsymbol{\nu}(t_{n+1}) - \boldsymbol{\nu}(t_n)}{\Delta t} =: \hat{\dot{\boldsymbol{\nu}}} \qquad (6.8) \end{aligned}$$

respectively with algorithmic parameters $\Theta_i \in \mathbb{R}$ $(i = 1, 2)$, and the linear approximation of the time-dependent external force within $[t_n, t_{n+1}]$ yields

$$\mathbf{f}(t_{n+\alpha}) \cong \mathbf{f}(t_n) + \frac{\mathbf{f}(t_{n+1}) - \mathbf{f}(t_n)}{\Delta t}[t_{n+\alpha} - t_n] =: \hat{\mathbf{f}} \tag{6.9}$$

Substituting Eq. (6.7), Eq. (6.8), and Eq. (6.9) into (6.6), the residual, $\mathbf{r}_{\mathrm{I}} \neq \mathbf{0}$, is defined as

$$\mathbf{r}_{\mathrm{I}} := \mathbf{M}\hat{\dot{\boldsymbol{\nu}}} + \hat{\mathbf{N}} - \hat{\mathbf{f}} \tag{6.10}$$

Let $\tau := t_{n+\alpha} - t_n = \alpha \Delta t$. By directly employing the ***classical (traditional) time weighted residual methodology***, we have

$$\frac{\int_0^{\Delta t} W \mathbf{r} d\tau}{\int_0^{\Delta t} W d\tau} = \frac{\int_0^{\Delta t} W \left[\mathbf{M}\hat{\dot{\boldsymbol{\nu}}} + \hat{\mathbf{N}} - \hat{\mathbf{f}}\right] d\tau}{\int_0^{\Delta t} W d\tau} \cong \mathbf{0} \tag{6.11}$$

or

$$\tilde{\mathbf{r}}_{\mathrm{I}} = \mathbf{M}\tilde{\dot{\boldsymbol{\nu}}} + \tilde{\mathbf{N}} - \tilde{\mathbf{f}} \cong \mathbf{0} \tag{6.12}$$

where the algorithmic unknowns are defined as shown in Eq. (4.12). The degenerated scalar polynomial weighting time field for the two-field form in nonlinear systems still takes the form

$$W(\Gamma) \cong w_0 + w_1 \Gamma + w_2 \Gamma^2 \quad (w_0 = 1); \tag{6.13}$$

Replacing

$$\begin{aligned}\frac{\int_0^{\Delta t}\frac{\tau}{\Delta t}W d\tau}{\int_0^{\Delta t}W d\tau} &= \frac{6+4w_1+3w_2}{12+6w_1+4w_2}, \text{ and}\\ \frac{\int_0^{\Delta t}\left(\frac{\tau}{\Delta t}\right)^2 W d\tau}{\int_0^{\Delta t}W d\tau} &= \frac{20+15w_1+12w_2}{60+30w_1+20w_2}\end{aligned} \tag{6.14}$$

with algorithmic parameters, W_1 and W_2, respectively, yields the algorithmic unknowns as follows

$$\tilde{\mathbf{N}} = \mathbf{N}_n + \Delta t W_1 \Theta_1 \dot{\mathbf{N}}_n + \Delta t W_2 \Theta_2 (\dot{\mathbf{N}}_{n+1} - \dot{\mathbf{N}}_n) \tag{6.15}$$

$$\tilde{\dot{\boldsymbol{\nu}}} = \frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t} \tag{6.16}$$

$$\tilde{\mathbf{f}} = \mathbf{f}_n + W_1(\mathbf{f}_{n+1} - \mathbf{f}_n) \tag{6.17}$$

The corresponding update for $\mathbf{N}$ is designed with new scalar algorithmic parameters $\theta_i \in \mathbb{R}$ $(i = 1, 2)$ as

$$\mathbf{N}_{n+1} = \mathbf{N}_n + \Delta t \theta_1 \dot{\mathbf{N}}_n + \Delta t \theta_2 (\dot{\mathbf{N}}_{n+1} - \dot{\mathbf{N}}_n) \tag{6.18}$$

The time discretization of the kinematic constraint is obtained from the updates for $\mathbf{q}$ and $\boldsymbol{\nu}$ as

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \Delta t \lambda_1 \boldsymbol{\nu}_n + \Delta t \lambda_2 \Delta \boldsymbol{\nu} \tag{6.19}$$

where $\Delta\boldsymbol{\nu} = \boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n$. Substituting Eq. (6.18) into Eq. (6.15), we can express the algorithmic $\mathbf{N}$ vector as

$$\tilde{\mathbf{N}} = \mathbf{N}_n + \frac{W_2\Theta_2}{\theta_2}(\mathbf{N}_{n+1} - \mathbf{N}_n) + \Delta t \left(W_1\Theta_1 - \frac{\theta_1}{\theta_2} W_2 \Theta_2 \right) \dot{\mathbf{N}}_n \tag{6.20}$$

Note that $\mathbf{N}_{n+1} = \mathbf{N}(\mathbf{q}_{n+1}, \boldsymbol{\nu}_{n+1}) = \mathbf{f}^{\text{int}}(\mathbf{q}_{n+1}) - \mathbf{f}^{\text{c}}(\mathbf{q}_{n+1}) - \mathbf{f}^{\text{diss}}(\boldsymbol{\nu}_{n+1}) \approx \mathbf{N}(t_{n+1})$. If the balance equation is given as Eq. (6.4), we have the velocity vector explicitly. In this case, we have

$$\tilde{\mathbf{r}}_{\text{I}} = \mathbf{M}\tilde{\dot{\boldsymbol{\nu}}} + \mathbf{C}\tilde{\boldsymbol{\nu}} + \tilde{\mathbf{f}}^{\text{pot}} - \tilde{\mathbf{f}} \cong \mathbf{0} \tag{6.21}$$

where the algorithmic potential force and velocity vectors are given by

$$\tilde{\mathbf{f}} = \mathbf{f}_n^{\text{pot}} + \frac{W_2\Theta_2}{\theta_2}(\mathbf{f}_{n+1}^{\text{pot}} - \mathbf{f}_n^{\text{pot}}) + \Delta t \left(W_1\Theta_1 - \frac{\theta_1}{\theta_2} W_2 \Theta_2 \right) \dot{\mathbf{f}}_n^{\text{pot}} \tag{6.22}$$

$$\tilde{\boldsymbol{\nu}} = \boldsymbol{\nu}_n + W_1 \Lambda_4 (\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n) \tag{6.23}$$

respectively.

Order of Time Accuracy

Define the local truncation errors from Eq. (6.21) and Eq. (6.19) as

$$\begin{aligned}
\boldsymbol{\tau}_1(\Delta t) &:= \mathbf{M}\left[\boldsymbol{\nu}(t_{n+1}) - \boldsymbol{\nu}(t_n)\right] + \Delta t\mathbf{C}\left[\boldsymbol{\nu}(t_n) + W_1\Lambda_4(\boldsymbol{\nu}(t_{n+1}) - \boldsymbol{\nu}(t_n))\right] \\
&\quad + \Delta t\left[\, \mathbf{f}^{\text{pot}}(t_n) + \frac{W_2\Theta_2}{\theta_2}(\mathbf{f}^{\text{pot}}(t_{n+1}) - \mathbf{f}^{\text{pot}}(t_n))\right. \\
&\quad \left. + \Delta t\left(W_1\Theta_1 - \frac{\theta_1}{\theta_2}W_2\Theta_2\right)\dot{\mathbf{f}}^{\text{pot}}(t_n)\,\right] - \Delta t\mathbf{f}(t_{n+W_1}) \equiv \mathbf{0} \qquad (6.24) \\
\boldsymbol{\tau}_2(\Delta t) &:= \mathbf{q}(t_{n+1}) - \mathbf{q}(t_n) - \Delta t\lambda_1\boldsymbol{\nu}(t_n) - \Delta t\lambda_2\left(\boldsymbol{\nu}(t_{n+1}) - \boldsymbol{\nu}(t_n)\right) \equiv \mathbf{0} \qquad (6.25)
\end{aligned}$$

Substituting the Taylor series expansions of $\mathbf{q}(t_{n+1})$, $\boldsymbol{\nu}(t_{n+1})$, $\mathbf{f}^{\text{pot}}(t_{n+1})$, and $\mathbf{f}(t_{n+W_1})$ at time t_n,

$$\begin{aligned}
\mathbf{q}(t_{n+1}) &= \mathbf{q}(t_n) + \Delta t\dot{\mathbf{q}}(t_n) + \frac{\Delta t^2}{2}\ddot{\mathbf{q}}(t_n) + \mathcal{O}(\Delta t^3) \qquad (6.26) \\
\boldsymbol{\nu}(t_{n+1}) &= \boldsymbol{\nu}(t_n) + \Delta t\dot{\boldsymbol{\nu}}(t_n) + \frac{\Delta t^2}{2}\ddot{\boldsymbol{\nu}}(t_n) + \mathcal{O}(\Delta t^3) \qquad (6.27) \\
\mathbf{f}^{\text{pot}}(t_{n+1}) &= \mathbf{f}^{\text{pot}}(t_n) + \Delta t\dot{\mathbf{f}}^{\text{pot}}(t_n) + \mathcal{O}(\Delta t^2) \qquad (6.28) \\
\mathbf{f}(t_{n+W_1}) &= \mathbf{f}(t_n) + \Delta tW_1\dot{\mathbf{f}}(t_n) + \mathcal{O}(\Delta t^2) \qquad (6.29)
\end{aligned}$$

into Eq. (6.24) and Eq. (6.25) yields

$$\begin{aligned}
\boldsymbol{\tau}_1(\Delta t) &= \Delta t\underbrace{\left[\mathbf{M}\dot{\boldsymbol{\nu}}(t_n) + \mathbf{C}\boldsymbol{\nu}(t_n) + \mathbf{f}^{\text{pot}}(t_n) - \mathbf{f}(t_n)\right]}_{=\mathbf{0}} \\
&\quad + \Delta t^2\left[\, \frac{1}{2}\mathbf{M}\ddot{\boldsymbol{\nu}}(t_n) + W_1\Lambda_4\mathbf{C}\dot{\boldsymbol{\nu}}(t_n)\right. \\
&\quad \left. + \left(\frac{W_2\Theta_2}{\theta_2}(1-\theta_1) + W_1\Theta_1\right)\dot{\mathbf{f}}^{\text{pot}}(t_n) - W_1\dot{\mathbf{f}}(t_n)\,\right] \\
&\quad + \mathcal{O}(\Delta t^3) \qquad (6.30)
\end{aligned}$$

and

$$\boldsymbol{\tau}_2(\Delta t) = \Delta t\left[\dot{\mathbf{q}}(t_n) - \lambda_1\boldsymbol{\nu}(t_n)\right] + \Delta t^2\left[\frac{1}{2}\ddot{\mathbf{q}}(t_n) - \lambda_2\dot{\boldsymbol{\nu}}(t_n)\right] + \mathcal{O}(\Delta t^3) \qquad (6.31)$$

Hence, for $\boldsymbol{\tau} := (\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)^T = \mathcal{O}(\Delta t^3)$ and $\|\boldsymbol{\tau}\| = \mathcal{O}(\Delta t^3)$, i.e., the second-order time accuracy of the algorithm, the following relations must be imposed:

$$\begin{aligned} &\frac{1}{2} = W_1\Lambda_4 = \frac{W_2\Theta_2}{\theta_2}(1-\theta_1) + W_1\Theta_1 = W_1 \\ &\lambda_1 = 1, \;\; \lambda_2 = \frac{1}{2} \end{aligned} \tag{6.32}$$

That is,

$$\begin{aligned} &W_1 = \frac{1}{2}, \;\; \Lambda_4 = 1, \;\; \Theta_1 = \theta_1 = 1 \\ &\lambda_1 = 1, \;\; \lambda_2 = \frac{1}{2} \end{aligned} \tag{6.33}$$

Notice that W_2, $\Theta_2 \neq 0$, and θ_2 are free parameters. We employ the relations from the linear case,

$$W_1 = \frac{3 + \rho_\infty^{\min} + \rho_\infty^{\max} - \rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \quad \text{and} \quad W_2\Theta_2 = \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \tag{6.34}$$

assuming $\Theta_1 = \Lambda_1 = 1$, $\Theta_2 = \lambda_2$ and $\theta_2 = \lambda_2$, the second-order time accurate member is given only by $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$; that is, $W_2\Theta_2 = 1/4$.

Time Level Consistency

The time weighted residual methodology is a technique of discretizing the balance equation in time so that the temporally discrete balance equation has the consistent algorithmic time level $t^* = t_{n+W_1}$. The algorithmic time level of the algorithmic discrete time-dependent external force vector is clearly t_{n+W_1}. The algorithmic time level of $\tilde{\mathbf{f}}^{\text{pot}}$ is investigated as follows:

$$\begin{aligned} \tilde{\mathbf{f}}^{\text{pot}} &= \mathbf{f}_n^{\text{pot}} + \frac{W_2\Theta_2}{\theta_2}(\mathbf{f}_{n+1}^{\text{pot}} - \mathbf{f}_n^{\text{pot}}) + \Delta t\left(W_1\Theta_1 - \frac{\theta_1}{\theta_2}W_2\Theta_2\right)\dot{\mathbf{f}}_n^{\text{pot}} \\ &= \mathbf{f}^{\text{pot}}(t_n) + \frac{W_2\Theta_2}{\theta_2}(\mathbf{f}^{\text{pot}}(t_{n+1}) - \mathbf{f}^{\text{pot}}(t_n)) \\ &\quad + \Delta t\left(W_1\Theta_1 - \frac{\theta_1}{\theta_2}W_2\Theta_2\right)\dot{\mathbf{f}}^{\text{pot}}(t_n) + \mathcal{O}(\Delta t^2) \\ &= \mathbf{f}^{\text{pot}}(t_n) + \Delta t\left[\frac{W_2\Theta_2}{\theta_2}(1-\theta_1) + W_1\Theta_1\right]\dot{\mathbf{f}}^{\text{pot}}(t_n) + \mathcal{O}(\Delta t^2) \end{aligned} \tag{6.35}$$

Therefore, the algorithmic time level of $\tilde{\mathbf{f}}^{\text{pot}}$ is t_{n+W_1} due to Eq. (6.32), i.e.,

$$\tilde{\mathbf{f}}^{\text{pot}} = \mathbf{f}^{\text{pot}}(t_n) + \Delta t W_1 \dot{\mathbf{f}}^{\text{pot}}(t_n) + \mathcal{O}(\Delta t^2) = \mathbf{f}^{\text{pot}}(t_{n+W_1}) \tag{6.36}$$

Also, it can be easily shown that the algorithmic time level of $\tilde{\boldsymbol{\nu}}$ is also t_{n+W_1}. Since the algorithmic time level of $\tilde{\dot{\boldsymbol{\nu}}}$ is $t_{n+1/2}$ or $W_1 = 1/2$ (fixed), the consistent algorithmic time level of the temporally discrete balance equation yields $t^* = t_{n+1/2}$:

$$\begin{aligned} \mathbf{0} &= \mathbf{M}\dot{\boldsymbol{\nu}}(t^*) + \mathbf{C}\boldsymbol{\nu}(t^*) + \mathbf{f}^{\text{pot}}(t^*) - \mathbf{f}(t^*) \\ &= \mathbf{M}\frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t} + \mathbf{C}\boldsymbol{\nu}_{n+1/2} + \mathbf{f}^{\text{pot}}_{n+1/2} - \mathbf{f}_{n+1/2} + \mathcal{O}(\Delta t^2) \end{aligned} \tag{6.37}$$

Notice that the consistent algorithmic time level implies the second-order time accuracy of the unknowns in the balance equation.

Algorithm 13

I-GSSSS Algorithms and Framework for Nonlinear Dynamical Systems (Two-Field Form): Option I

Integrator:

$$\mathbf{M}\frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t} + \tilde{\mathbf{N}} = \tilde{\mathbf{f}}$$

where

$$\begin{aligned} \tilde{\mathbf{N}} &= \mathbf{N}_n + \frac{W_2\Theta_2}{\theta_2}(\mathbf{N}_{n+1} - \mathbf{N}_n) + \Delta t \left(W_1\Theta_1 - \frac{\theta_1}{\theta_2} W_2\Theta_2 \right) \dot{\mathbf{N}}_n \\ \tilde{\mathbf{f}} &= \mathbf{f}_{n+W_1} \;\; or \;\; \mathbf{f}(t_{n+W_1}) \end{aligned}$$

Updates:

$$\begin{aligned} \mathbf{q}_{n+1} &= \mathbf{q}_n + \Delta t \lambda_1 \boldsymbol{\nu}_n + \Delta t \lambda_2 \Delta \boldsymbol{\nu} \\ \boldsymbol{\nu}_{n+1} &= \boldsymbol{\nu}_n + \Delta \boldsymbol{\nu} \end{aligned}$$

Initial conditions:

$$\begin{aligned} \mathbf{q}(t_0) &= \mathbf{q}_0 \\ \boldsymbol{\nu}(t_0) &= \boldsymbol{\nu}_0 \end{aligned}$$

Remark 14 (Algorithm 13)

1. *If the balance equation is given as Eq.* (6.4)*, Algorithm 13 may be written as the following:*

   Integrator:

$$\mathbf{M}\frac{\boldsymbol{\nu}_{n+1}-\boldsymbol{\nu}_n}{\Delta t}+\mathbf{C}\tilde{\boldsymbol{\nu}}+\tilde{\mathbf{f}}^{\mathrm{pot}}=\tilde{\mathbf{f}}$$

   where

$$\begin{aligned}\tilde{\mathbf{f}}^{\mathrm{pot}} &= \mathbf{f}_n^{\mathrm{pot}}+\frac{W_2\Theta_2}{\theta_2}(\mathbf{f}_{n+1}^{\mathrm{pot}}-\mathbf{f}_n^{\mathrm{pot}})+\Delta t\left(W_1\Theta_1-\frac{\theta_1}{\theta_2}W_2\Theta_2\right)\dot{\mathbf{f}}_n^{\mathrm{pot}}\\ \tilde{\boldsymbol{\nu}} &= \boldsymbol{\nu}_n+W_1\Lambda_4(\boldsymbol{\nu}_{n+1}-\boldsymbol{\nu}_n)\\ \tilde{\mathbf{f}} &= \mathbf{f}_{n+W_1}\ \ or\ \ \mathbf{f}(t_{n+W_1})\end{aligned}$$

   Updates:

$$\begin{aligned}\mathbf{q}_{n+1}&=\mathbf{q}_n+\Delta t\lambda_1\boldsymbol{\nu}_n+\Delta t\lambda_2\Delta\boldsymbol{\nu}\\ \boldsymbol{\nu}_{n+1}&=\boldsymbol{\nu}_n+\Delta\boldsymbol{\nu}\end{aligned}$$

   Note that $\mathbf{f}_{n+1}^{\mathrm{pot}}=\mathbf{f}^{\mathrm{pot}}(\mathbf{q}_{n+1})=\mathbf{f}^{\mathrm{int}}(\mathbf{q}_{n+1})-\mathbf{f}^{\mathrm{c}}(\mathbf{q}_{n+1})=\nabla\mathscr{U}(\mathbf{q}_{n+1})$*. In linear dynamical systems, we have* $\mathbf{f}^{\mathrm{pot}}(\mathbf{q})\cong\mathbf{K}\mathbf{q}$*; hence, the algorithmic potential force vector yields*

$$\tilde{\mathbf{f}}^{\mathrm{pot}}\cong\mathbf{K}\left[\mathbf{q}_n+\frac{W_2\Theta_2}{\theta_2}(\mathbf{q}_{n+1}-\mathbf{q}_n)+\Delta t\left(W_1\Theta_1-\frac{\theta_1}{\theta_2}W_2\Theta_2\right)\boldsymbol{\nu}_n\right]\tag{6.38}$$

   By letting $\Theta_i=\Lambda_i$ *and* $\theta_i=\lambda_i$ *for* $i=1,2$*, we have*

$$\tilde{\mathbf{f}}^{\mathrm{pot}}\cong\mathbf{K}\left[\mathbf{q}_n+\frac{W_2\Lambda_2}{\lambda_2}(\mathbf{q}_{n+1}-\mathbf{q}_n)+\Delta t\left(W_1\Lambda_1-\frac{\lambda_1}{\lambda_2}W_2\Lambda_2\right)\boldsymbol{\nu}_n\right]=\mathbf{K}\tilde{\mathbf{q}}\tag{6.39}$$

   and Algorithm 13 recovers Algorithm 1 as expected. Since Algorithm 13 is an extension based on Algorithm 1 to nonlinear dynamical systems, the assumptions $\Theta_i=\Lambda_i$ *and* $\theta_i=\lambda_i$ *for* $i=1,2$ *are consistent and appropriate.*

2. *Imposing the conditions for the second-order time accuracy, i.e., Eq.* (6.33)*,*

we get

$$\begin{aligned}
&\mathbf{M}\frac{\boldsymbol{\nu}_{n+1}-\boldsymbol{\nu}_n}{\Delta t}+\mathbf{C}\frac{\boldsymbol{\nu}_{n+1}+\boldsymbol{\nu}_n}{2}\\
&+\mathbf{f}_n^{\mathrm{pot}}+\frac{W_2\Theta_2}{\theta_2}(\mathbf{f}_{n+1}^{\mathrm{pot}}-\mathbf{f}_n^{\mathrm{pot}})+\Delta t\left(\frac{1}{2}-\frac{W_2\Theta_2}{\theta_2}\right)\dot{\mathbf{f}}_n^{\mathrm{pot}}=\mathbf{f}(t_{n+1/2})\\
&\frac{\mathbf{q}_{n+1}-\mathbf{q}_n}{\Delta t}=\frac{\boldsymbol{\nu}_{n+1}+\boldsymbol{\nu}_n}{2}
\end{aligned}\tag{6.40}$$

Imposing $W_2\Theta_2=1/4$ *due to* $\rho_\infty^{\min}=\rho_\infty^{\max}=1$ *in Eq.* (6.34) *and selecting* $\theta_2=1/2$*, we recover the two-field form trapezoidal method:*

$$\begin{aligned}
&\mathbf{M}\frac{\boldsymbol{\nu}_{n+1}-\boldsymbol{\nu}_n}{\Delta t}+\mathbf{C}\frac{\boldsymbol{\nu}_{n+1}+\boldsymbol{\nu}_n}{2}+\frac{\mathbf{f}_{n+1}^{\mathrm{pot}}+\mathbf{f}_n^{\mathrm{pot}}}{2}=\mathbf{f}(t_{n+1/2})\\
&\frac{\mathbf{q}_{n+1}-\mathbf{q}_n}{\Delta t}=\frac{\boldsymbol{\nu}_{n+1}+\boldsymbol{\nu}_n}{2}
\end{aligned}\tag{6.41}$$

3. ***Application to the Nonlinear Elastodynamical Systems:*** *Application of Algorithm 13 for the initial-value problem in the sense of the two-field form in the nonlinear elastodynamical system with* $\boldsymbol{\nu}^j=\dot{\mathbf{q}}^j$ *yields*

$$\begin{aligned}
&\sum_{j=1}^{n_{\mathrm{node}}}M_{ij}\frac{\boldsymbol{\nu}_{n+1}^j-\boldsymbol{\nu}_n^j}{\Delta t}+\tilde{\mathbf{F}}_i^{\mathrm{int}^h}\\
&=\int_{\mathcal{B}}N_i\rho_0\mathbf{B}(t_{n+W_1})dV+\int_{\partial\mathcal{B}_\sigma}N_i\bar{\mathbf{T}}(t_{n+W_1})dA
\end{aligned}\tag{6.42}$$

with the updates and the initial conditions (given in Algorithm 13), where $\tilde{\mathbf{F}}^{\mathrm{int}^h}$ *denotes the algorithmic discrete internal force vector defined by*

$$\tilde{\mathbf{F}}^{\mathrm{int}^h}=\mathbf{F}_n^{\mathrm{int}^h}+\frac{W_2\Theta_2}{\theta_2}(\mathbf{F}_{n+1}^{\mathrm{int}^h}-\mathbf{F}_n^{\mathrm{int}^h})+\Delta t\left(W_1\Theta_1-\frac{\theta_1}{\theta_2}W_2\Theta_2\right)\dot{\mathbf{F}}_n^{\mathrm{int}^h}\tag{6.43}$$

Note that $\mathbf{F}_n^{\mathrm{int}^h}\approx\mathbf{F}^{\mathrm{int}^h}(t_n)$ *and* $\mathbf{F}_{n+1}^{\mathrm{int}^h}\approx\mathbf{F}^{\mathrm{int}^h}(t_{n+1})$*, and* $\mathbf{F}_n^{\mathrm{int}^h}$ *is given as*

$$\begin{aligned}
\mathbf{F}_{n\,i}^{\mathrm{int}}&=\int_{\mathcal{B}}\mathbf{F}_n^h\cdot\mathbf{S}_n^h\,\mathrm{GRAD}\,N_i\,dV\\
&=\sum_{j=1}^{n_{\mathrm{node}}}\int_{\mathcal{B}}\mathrm{GRAD}\,N_i\cdot\mathbf{S}_n^h\mathrm{GRAD}\,N_j dV\,\mathbf{q}_n^j
\end{aligned}\tag{6.44}$$

where $\mathbf{F}_n^h \approx \mathbf{F}^h(\mathbf{X}, t_n)$ *denotes the discrete deformation gradient tensor field approximated at time* t_n *given by*

$$\mathbf{F}_n^h = \mathbf{I} + \sum_{i=1}^{n_{\text{node}}} \mathbf{q}_n^i \otimes \text{GRAD}\, N_i(\mathbf{X}) \tag{6.45}$$

and $\mathbf{S}_n^h \approx \mathbf{S}^h(\mathbf{X}, t_n)$ *denotes the discrete second Piola-Kirchhoff stress tensor field approximated at time* t_n *given by* $\mathbf{S}_n^h = \hat{\mathbf{S}}^h(\mathbf{C}_n^h) = \bar{\mathbf{S}}^h(\mathbf{F}_n^h)$ *where* $\mathbf{C}_n^h = \mathbf{F}_n^{h\,T}\mathbf{F}_n^h$. *Recall that* $\hat{\mathbf{S}}^h$ *and* $\bar{\mathbf{S}}^h$ *are the response functions for* $\mathbf{S}^h$ *in terms of the discrete right Cauchy-Green strain tensor field* $\mathbf{C}^h$ *and the discrete deformation gradient tensor field* $\mathbf{F}^h$, *respectively. Hence, the second-order time accurate trapezoidal rule for this application is:*

$$\begin{aligned}
&\sum_{j=1}^{n_{\text{node}}} M_{ij} \frac{\boldsymbol{\nu}_{n+1}^j - \boldsymbol{\nu}_n^j}{\Delta t} \\
&+ \int_{\mathcal{B}} \frac{\mathbf{F}_n^h \cdot \mathbf{S}_n^h + \mathbf{F}_{n+1}^h \cdot \mathbf{S}_{n+1}^h}{2} \text{GRAD}\, N_i \, dV \\
&= \int_{\mathcal{B}} N_i \rho_0 \mathbf{B}(t_{n+1/2}) dV + \int_{\partial \mathcal{B}_\sigma} N_i \bar{\mathbf{T}}(t_{n+1/2}) dA \\
&\frac{\mathbf{q}_{n+1}^i - \mathbf{q}_n^i}{\Delta t} = \frac{\boldsymbol{\nu}_{n+1}^i + \boldsymbol{\nu}_n^i}{2}
\end{aligned} \tag{6.46}$$

with the initial conditions. Notice the algorithmic time level of Eq. (6.46) *is* $t^* = t_{n+1/2}$.

4. *The framework based upon Option I via the classical (traditional) time weighted residual methodology does not provide attractive time-stepping algorithms which conserve total angular momentum or total energy within a time step for any spectral conditions, although the total linear momentum within the time step is exactly conserved for any spectral conditions in the conservative system. To overcome this difficulty, a new and novel* ***normalized time weighted residual methodology*** *is presented next.*

6.1.2 Option II: Symplectic-Momentum Conserving Algorithms and Designs via Normalized Time Weighted Residual Methodology

Consider the nonlinear two-field form system,

$$\boxed{\begin{array}{l} \textbf{\textit{Balance Equation:}} \\ \mathbf{M}\dot{\boldsymbol{\nu}}(t) + \mathbf{N}(\mathbf{q}, \boldsymbol{\nu}) = \mathbf{f}(t) \\ \textbf{\textit{Kinematic constraint:}} \\ \dot{\mathbf{q}}(t) = \boldsymbol{\nu}(t) \quad \forall t \in \mathbb{I} \\ \textbf{\textit{Initial conditions:}} \\ \mathbf{q}(t_0) = \mathbf{q}_0 \\ \boldsymbol{\nu}(t_0) = \boldsymbol{\nu}_0 \end{array}} \tag{6.47}$$

From the balance equation, we have

$$\mathbf{M}\dot{\boldsymbol{\nu}}(t) + \mathbf{N}(\mathbf{q}(t), \boldsymbol{\nu}(t)) - \mathbf{f}(t) = \mathbf{0} \ \ \forall t \in \mathbb{I} \tag{6.48}$$

Unlike *Option I* in the previous subsection, we specially approximate $\mathbf{N}$ by approximating the variables $\mathbf{q}$ and $\boldsymbol{\nu}$. The asymptotic series expansions of the dependent variables, $\mathbf{q}$, $\boldsymbol{\nu}$, and $\dot{\boldsymbol{\nu}}$, about time $t = t_{n+\alpha} := (1-\alpha)t_n + \alpha t_{n+1} \in [t_n, t_{n+1}]$ for $\alpha \in [0,1]$ yield

$$\begin{aligned} \mathbf{q}(t_{n+\alpha}) &\cong \mathbf{q}(t_n) + \Lambda_1 \dot{\mathbf{q}}(t_n)[t_{n+\alpha} - t_n] \\ &\quad + \Lambda_2 \frac{\dot{\mathbf{q}}(t_{n+1}) - \dot{\mathbf{q}}(t_n)}{\Delta t}[t_{n+\alpha} - t_n]^2 \qquad (6.49) \\ \boldsymbol{\nu}(t_{n+\alpha}) &\cong \boldsymbol{\nu}(t_n) + \Lambda_4 \frac{\boldsymbol{\nu}(t_{n+1}) - \boldsymbol{\nu}(t_n)}{\Delta t}[t_{n+\alpha} - t_n] =: \hat{\boldsymbol{\nu}} \qquad (6.50) \\ \dot{\boldsymbol{\nu}}(t_{n+\alpha}) &\cong \frac{\boldsymbol{\nu}(t_{n+1}) - \boldsymbol{\nu}(t_n)}{\Delta t} =: \hat{\dot{\boldsymbol{\nu}}} \qquad (6.51) \end{aligned}$$

respectively with algorithmic parameters $\Lambda_i \in \mathbb{R}$ $(i = 1, 2, 3)$. Because of the kinematic constraints at time t_n and t_{n+1}, i.e., $\dot{\mathbf{q}}(t_n) = \boldsymbol{\nu}(t_n)$ and $\dot{\mathbf{q}}(t_{n+1}) =$

$\boldsymbol{\nu}(t_{n+1})$, respectively, Eq. (6.49) may be written as

$$\begin{aligned}\mathbf{q}(t_{n+\alpha}) \cong & \mathbf{q}(t_n) + \Lambda_1 \boldsymbol{\nu}(t_n)[t_{n+\alpha} - t_n] \\ & + \Lambda_2 \frac{\boldsymbol{\nu}(t_{n+1}) - \boldsymbol{\nu}(t_n)}{\Delta t}[t_{n+\alpha} - t_n]^2 =: \hat{\mathbf{q}}\end{aligned} \tag{6.52}$$

For the time-dependent external force vector, we linearly approximate within a time step $[t_n, t_{n+1}]$ as

$$\mathbf{f}(t_{n+\alpha}) \cong \mathbf{f}(t_n) + \frac{\mathbf{f}(t_{n+1}) - \mathbf{f}(t_n)}{\Delta t}[t_{n+\alpha} - t_n] =: \hat{\mathbf{f}} \tag{6.53}$$

Substituting Eq. (6.50) - Eq. (6.52) into (6.53), the residual, $\mathbf{r}_{\mathrm{II}}$, is defined as

$$\mathbf{r}_{\mathrm{II}} := \mathbf{M}\hat{\boldsymbol{\nu}} + \mathbf{N}(\hat{\mathbf{q}}, \hat{\boldsymbol{\nu}}) - \hat{\mathbf{f}} \tag{6.54}$$

The procedure for the normalized time weighted residual methodology [7, 8] is as follows. We define the algorithmic unknowns independently in the balance equation with the degenerated scalar polynomial weighting time field W defined in Eq. (6.13) and set the temporally discretized balance equation to be zero as follows:

$$\mathbf{M}\frac{\int_0^{\Delta t} W\hat{\boldsymbol{\nu}} d\tau}{\int_0^{\Delta t} W d\tau} + \mathbf{N}(\frac{\int_0^{\Delta t} W\hat{\mathbf{q}} d\tau}{\int_0^{\Delta t} W d\tau}, \frac{\int_0^{\Delta t} W\hat{\boldsymbol{\nu}} d\tau}{\int_0^{\Delta t} W d\tau}) - \frac{\int_0^{\Delta t} W\hat{\mathbf{f}} d\tau}{\int_0^{\Delta t} W d\tau} \cong \mathbf{0} \tag{6.55}$$

or

$$\mathbf{M}\tilde{\dot{\boldsymbol{\nu}}} + \mathbf{N}(\tilde{\mathbf{q}}, \tilde{\boldsymbol{\nu}}) - \tilde{\mathbf{f}} \cong \mathbf{0} \tag{6.56}$$

where

$$\begin{aligned}\tilde{\mathbf{q}} &= \mathbf{q}_n + \Delta t W_1 \Lambda_1 \boldsymbol{\nu}_n + \Delta t W_2 \Lambda_2 \Delta\boldsymbol{\nu} \\ \tilde{\boldsymbol{\nu}} &= \boldsymbol{\nu}_n + W_1 \Lambda_4 \Delta\boldsymbol{\nu} \\ \tilde{\dot{\mathbf{v}}} &= \Delta\boldsymbol{\nu}/\Delta t \\ \tilde{\mathbf{f}} &= \mathbf{f}_n + W_1(\mathbf{f}_{n+1} - \mathbf{f}_n)\end{aligned} \tag{6.57}$$

respectively ($\Delta\boldsymbol{\nu} := \boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n$). The corresponding updates are obtained as

$$\begin{aligned}\mathbf{q}_{n+1} &= \mathbf{q}_n + \Delta t \lambda_1 \boldsymbol{\nu}_n + \Delta t \lambda_2 \Delta\boldsymbol{\nu} \\ \boldsymbol{\nu}_{n+1} &= \boldsymbol{\nu}_n + \Delta\boldsymbol{\nu}\end{aligned} \tag{6.58}$$

which can be considered as the discrete form of the kinematic constraint, i.e.,

$$\frac{\mathbf{q}_{n+1}-\mathbf{q}_n}{\Delta t}=\lambda_1\boldsymbol{\nu}_n+\lambda_2(\boldsymbol{\nu}_{n+1}-\boldsymbol{\nu}_n) \tag{6.59}$$

Notice that the algorithmic configuration $\tilde{\mathbf{q}}$ yields

$$\tilde{\mathbf{q}}=\mathbf{q}_n+\frac{W_2\Lambda_2}{\lambda_2}(\mathbf{q}_{n+1}-\mathbf{q}_n)+\Delta t\left(W_1\Lambda_1-\frac{\lambda_1}{\lambda_2}W_2\Lambda_2\right)\boldsymbol{\nu}_n \tag{6.60}$$

Algorithm 14

I-GSSSS Algorithms and Designs for Nonlinear Dynamical Systems (Two-Field Form): Option II

Integrator:

$$\mathbf{M}\frac{\boldsymbol{\nu}_{n+1}-\boldsymbol{\nu}_n}{\Delta t}+\mathbf{N}(\tilde{\mathbf{q}},\tilde{\boldsymbol{\nu}})=\tilde{\mathbf{f}}$$

where

$$\begin{aligned}\tilde{\mathbf{q}}&=\mathbf{q}_n+\Delta tW_1\Lambda_1\boldsymbol{\nu}_n+\Delta tW_2\Lambda_2\Delta\boldsymbol{\nu}\\ \tilde{\boldsymbol{\nu}}&=\boldsymbol{\nu}_n+W_1\Lambda_4\Delta\boldsymbol{\nu}\\ \tilde{\mathbf{f}}&=\mathbf{f}_{n+W_1}\ \ or\ \ \mathbf{f}(t_{n+W_1})\end{aligned}$$

Updates:

$$\begin{aligned}\mathbf{q}_{n+1}&=\mathbf{q}_n+\Delta t\lambda_1\boldsymbol{\nu}_n+\Delta t\lambda_2\Delta\boldsymbol{\nu}\\ \boldsymbol{\nu}_{n+1}&=\boldsymbol{\nu}_n+\Delta\boldsymbol{\nu}\end{aligned}$$

Initial conditions:

$$\begin{aligned}\mathbf{q}(t_0)&=\mathbf{q}_0\\ \boldsymbol{\nu}(t_0)&=\boldsymbol{\nu}_0\end{aligned}$$

Remark 15 (Algorithm 14)

1. *The integrator for a standard case can be written as*

$$\mathbf{M}\frac{\boldsymbol{\nu}_{n+1}-\boldsymbol{\nu}_n}{\Delta t}+\mathbf{f}^{\text{int}}(\tilde{\mathbf{q}},\tilde{\boldsymbol{\nu}})=\mathbf{f}^{\text{ext}}(\tilde{\mathbf{q}},\tilde{\boldsymbol{\nu}},t_{n+W_1})$$

Also, the integrator for a special case (Eq. (6.4)) yields

$$\mathbf{M}\frac{\boldsymbol{\nu}_{n+1}-\boldsymbol{\nu}_n}{\Delta t}+\mathbf{C}\tilde{\boldsymbol{\nu}}+\mathbf{f}^{\text{pot}}(\tilde{\mathbf{q}})=\mathbf{f}(t_{n+W_1}) \tag{6.61}$$

In linear dynamical systems, Algorithm 14 recovers Algorithm 1 since $\mathbf{f}^{\text{pot}}(\tilde{\mathbf{q}}) = \nabla\mathscr{U}(\tilde{\mathbf{q}}) \cong \mathbf{K}\tilde{\mathbf{q}}$. *Note also that for the linear case both the classical and normalized time weighted residual methodology yield identical results. They differ in the outcomes for nonlinear dynamical applications.*

2. ***Time level consistency:*** *Since we use the same algorithmic parameters from Algorithm 1, impose* $\lambda_1 = \Lambda_1 = \Lambda_4 = 1$ *and* $\lambda_2 = 1/2$ *in Eq.* (6.60) *and* $\tilde{\boldsymbol{\nu}}$ *in Eq.* (6.57)*; then, we get*

$$\begin{aligned} \tilde{\mathbf{q}} &= \mathbf{q}_n + 2W_2\Lambda_2(\mathbf{q}_{n+1} - \mathbf{q}_n) + \Delta t\,(W_1 - 2W_2\Lambda_2)\,\boldsymbol{\nu}_n \qquad (6.62)\\ \tilde{\boldsymbol{\nu}} &= (1 - W_1)\boldsymbol{\nu}_n + W_1\boldsymbol{\nu}_{n+1} \qquad (6.63) \end{aligned}$$

Hence, the algorithmic time level is shown to be $t^* = t_{n+1/2}$ *from the algorithmic time level consistency of the balance equation*

$$\begin{aligned} \mathbf{0} &= \mathbf{M}\dot{\boldsymbol{\nu}}(t^*) + \mathbf{N}(\mathbf{q}(t^*), \boldsymbol{\nu}(t^*)) - \mathbf{f}(t^*) \\ &= \mathbf{M}\frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t} + \mathbf{N}(\mathbf{q}_{n+W_1}, \boldsymbol{\nu}_{n+W_1}) - \mathbf{f}_{n+W_1} + \mathcal{O}(\Delta t^p) \end{aligned} \qquad (6.64)$$

where $p = 2$ *if* $W_1 = 1/2$*; otherwise,* $p = 1$*. There exists only one second-order time accurate algorithm in Algorithm 14. Selecting the parameters as*

$$\begin{aligned} &W_1 = \tfrac{1}{2},\ \ W_2\Lambda_2 = \tfrac{1}{4},\ \ \lambda_1 = \Lambda_1 = \Lambda_4 = 1,\ \ \textit{and}\ \ \lambda_2 = \tfrac{1}{2} \\ &(\rho_\infty^{\min} = \rho_\infty^{\max} = 1) \end{aligned} \qquad (6.65)$$

Algorithm 14 yields

$$\begin{aligned} &\mathbf{M}\frac{\dot{\mathbf{q}}_{n+1} - \dot{\mathbf{q}}_n}{\Delta t} + \mathbf{N}(\frac{\mathbf{q}_{n+1} + \mathbf{q}_n}{2}, \frac{\boldsymbol{\nu}_{n+1} + \boldsymbol{\nu}_n}{2}) = \mathbf{f}(t_{n+1/2}) \\ &\frac{\boldsymbol{\nu}_{n+1} + \boldsymbol{\nu}_n}{2} = \frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t} \end{aligned} \qquad (6.66)$$

In a conservative system, it reduces to the classical ***midpoint rule in the two-field form****,*

$$\begin{aligned} &\mathbf{M}\frac{\dot{\mathbf{q}}_{n+1} - \dot{\mathbf{q}}_n}{\Delta t} + \nabla\mathscr{U}(\frac{\mathbf{q}_{n+1} + \mathbf{q}_n}{2}) = \mathbf{0} \\ &\frac{\boldsymbol{\nu}_{n+1} + \boldsymbol{\nu}_n}{2} = \frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t} \end{aligned} \qquad (6.67)$$

Setting $\Lambda_2 = \lambda_2 = \frac{1}{2}$ *in Eq.* (6.65), W_1 *and* W_2 *defined in Eq.* (6.14) *are explicitly given as*

$$W_1 = W_2 = \tfrac{1}{2} \Longleftrightarrow w_1 = -5 \;\; and \;\; w_2 = 5 \tag{6.68}$$

The parameters w_1 *and* w_2 *remain constant during the simulation. It is well-known that the midpoint rule in the two-field form is the symplectic-momentum conserving scheme. Note that the normalized time weighted residual methodology in the sense of Option II naturally leads to the symplectic-momentum conserving algorithm.*

3. ***Application to the Nonlinear Elastodynamical Systems:*** *Application of Algorithm 13 for the initial-value problem in the sense of the two-field form in the nonlinear elastodynamical system with* $\boldsymbol{\nu}^j = \dot{\mathbf{q}}^j$ *yields*

$$\begin{aligned} &\sum_{j=1}^{n_{\text{node}}} M_{ij} \frac{\boldsymbol{\nu}^j_{n+1} - \boldsymbol{\nu}^j_n}{\Delta t} + \mathbf{F}^{\text{int}^h}_i(\tilde{\mathbf{q}}) \\ &= \int_{\mathcal{B}} N_i \rho_0 \mathbf{B}(t_{n+W_1}) dV + \int_{\partial \mathcal{B}_\sigma} N_i \bar{\mathbf{T}}(t_{n+W_1}) dA \end{aligned} \tag{6.69}$$

with the updates and the initial conditions, (given in Algorithm 14), where $\mathbf{F}^{\text{int}^h}(\tilde{\mathbf{q}})$ *denotes the algorithmic discrete internal force vector evaluated by* $\tilde{\mathbf{q}}$*, i.e.,*

$$\begin{aligned} \mathbf{F}^{\text{int}}_{n\,i} &= \int_{\mathcal{B}} \tilde{\mathbf{F}}^h \cdot \bar{\mathbf{S}}^h(\tilde{\mathbf{F}}^h) \; \text{GRAD} \; N_i \, dV \\ &= \sum_{j=1}^{n_{\text{node}}} \int_{\mathcal{B}} \text{GRAD}\, N_i \cdot \bar{\mathbf{S}}^h(\tilde{\mathbf{F}}^h) \text{GRAD}\, N_j dV \tilde{\mathbf{q}}^j \end{aligned} \tag{6.70}$$

in which $\tilde{\mathbf{F}}^h_n$ *denotes the algorithmic discrete deformation gradient tensor field given by*

$$\tilde{\mathbf{F}}^h = \mathbf{I} + \sum_{i=1}^{n_{\text{node}}} \tilde{\mathbf{q}}^i \otimes \text{GRAD}\, N_i(\mathbf{X}) \tag{6.71}$$

Therefore, in this framework, the algorithmic discrete second Piola-Kirchhoff stress tensor field, $\tilde{\mathbf{S}}^h$*, and algorithmic discrete right Cauchy-Green strain*

tensor field, $\tilde{\mathbf{C}}^h$, are deinfed as

$$\tilde{\mathbf{S}}^h := \hat{\mathbf{S}}^h(\tilde{\mathbf{C}}^h) = \bar{\mathbf{S}}^h(\tilde{\mathbf{F}}^h) \tag{6.72}$$

$$\tilde{\mathbf{C}}^h := \tilde{\mathbf{F}}^{h\,T}\tilde{\mathbf{F}}^h \tag{6.73}$$

respectively. It is also important to note $\tilde{\mathbf{S}}^h$ can be expressed via the discrete strain energy function as $\tilde{\mathbf{S}}^h = 2D\hat{W}^h(\tilde{\mathbf{C}}^h)$ where D denotes the derivative at $\tilde{\mathbf{C}}^h$. Hence, the second-order time accurate, symplectic midpoint rule for this application takes the form,

$$\begin{aligned}
&\sum_{j=1}^{n_{\text{node}}} M_{ij}\frac{\boldsymbol{\nu}^j_{n+1}-\boldsymbol{\nu}^j_n}{\Delta t} + \int_{\mathcal{B}} \mathbf{F}^h_{n+1/2}\cdot\bar{\mathbf{S}}^h(\mathbf{F}^h_{n+1/2})\,\text{GRAD}\,N_i\,dV \\
&= \int_{\mathcal{B}} N_i\rho_0\mathbf{B}(t_{n+1/2})dV + \int_{\partial\mathcal{B}_\sigma} N_i\bar{\mathbf{T}}(t_{n+1/2})dA \\
&\frac{\mathbf{q}^i_{n+1}-\mathbf{q}^i_n}{\Delta t} = \frac{\boldsymbol{\nu}^i_{n+1}+\boldsymbol{\nu}^i_n}{2}
\end{aligned} \tag{6.74}$$

with the initial conditions, where

$$\mathbf{F}^h_{n+1/2} = \mathbf{I} + \sum_{i=1}^{n_{\text{node}}} \mathbf{q}^i_{n+1/2}\otimes\text{GRAD}\,N_i(\mathbf{X}) \tag{6.75}$$

Notice the algorithmic time level of Eq. (6.74) *is consistently $t^* = t_{n+1/2}$.*

6.2 Nonlinear I-GSSSS Algorithms and Framework in Single-Field Form

The standard equation of motion in the single field form in nonlinear dynamical systems may be written as the following initial-value problem,

$$\boxed{\begin{aligned}
&\textbf{\textit{Balance Equation:}}\\
&\mathbf{M}\ddot{\mathbf{q}} + \mathbf{f}^{\text{int}}(\mathbf{q},\dot{\mathbf{q}}) = \mathbf{f}^{\text{ext}}(\mathbf{q},\dot{\mathbf{q}},t)\\
&\textbf{\textit{Initial conditions:}}\\
&\mathbf{q}(t_0) = \mathbf{q}_0\\
&\dot{\mathbf{q}}(t_0) = \dot{\mathbf{q}}_0
\end{aligned}} \tag{6.76}$$

where $\mathbf{f}^{\text{int}}(\mathbf{q},\dot{\mathbf{q}}) : TQ \to \mathbb{R}^{3N}$ and $\mathbf{f}^{\text{ext}}(\mathbf{q},\dot{\mathbf{q}},t) : TQ \times \mathbb{I} \to \mathbb{R}^{3N}$ are the internal and external forces, respectively. If $\mathbf{f}^{\text{int}}(\mathbf{q}) := \nabla \mathscr{U}^{\text{int}}(\mathbf{q}) : Q \to \mathbb{R}^{3N}$ and $\mathbf{f}^{\text{ext}}(\mathbf{q},\dot{\mathbf{q}},t) = \mathbf{f}(t) + \mathbf{f}^{\text{diss}}(\dot{\mathbf{q}}) + \mathbf{f}^{\text{c}}(\mathbf{q})$ where $\mathbf{f}(t) : \mathbb{I} \to \mathbb{R}^{3N}$, $\mathbf{f}^{\text{diss}}(\dot{\mathbf{q}}) : T_{\mathbf{q}}Q \to \mathbb{R}^{3N}$, and $\mathbf{f}^{\text{c}}(\mathbf{q}) : Q \to \mathbb{R}^{3N}$ are the time-dependent external force, the dissipative force, and the external conservative force, respectively, the balance equation becomes

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{f}^{\text{int}}(\mathbf{q}) = \mathbf{f}(t) + \mathbf{f}^{\text{diss}}(\dot{\mathbf{q}}) + \mathbf{f}^{\text{c}}(\mathbf{q}) \tag{6.77}$$

Note that the external conservative force is defined from the external potential energy functions as $\mathbf{f}^{\text{c}}(\mathbf{q}) := -\nabla \mathscr{U}^{\text{ext}}(\mathbf{q})$. Defining $\mathbf{N}(\mathbf{q},\dot{\mathbf{q}}) := \mathbf{f}^{\text{int}}(\mathbf{q}) - \mathbf{f}^{\text{c}}(\mathbf{q}) - \mathbf{f}^{\text{diss}}(\dot{\mathbf{q}})$, Eq. (6.77) can be written as

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{N}(\mathbf{q},\dot{\mathbf{q}}) = \mathbf{f}(t) \tag{6.78}$$

If the dissipative force is given linearly as $\mathbf{f}^{\text{diss}}(\dot{\mathbf{q}}) = -\mathbf{C}\dot{\mathbf{q}}$, Eq. (6.78) can be written as

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{f}^{\text{pot}}(\mathbf{q}) = \mathbf{f}(t) \tag{6.79}$$

where $\mathbf{f}^{\text{pot}}(\mathbf{q}) := \mathbf{f}^{\text{int}}(\mathbf{q}) - \mathbf{f}^{\text{c}}(\mathbf{q}) = \nabla \mathscr{U}$ is the potential force.

6.2.1 Option I: Algorithms and Designs Via Classical Time Weighted Residual Methodology

Following the similar procedure of the nonlinear extension by *Option I* in the two-field form (see Subsection 6.1.1), we consider

$$\boxed{\begin{array}{l} \textbf{\textit{Balance Equation:}} \\ \mathbf{M}\ddot{\mathbf{q}}(t) + \mathbf{N}(t) = \mathbf{f}(t) \\ \textbf{\textit{Initial conditions:}} \\ \mathbf{q}(t_0) = \mathbf{q}_0 \\ \dot{\mathbf{q}}(t_0) = \dot{\mathbf{q}}_0 \end{array}} \tag{6.80}$$

where $\mathbf{N}(t) \equiv \mathbf{N}(\mathbf{q}(t),\dot{\mathbf{q}}(t))$. From the balance equation in Eq. (6.80), we get

$$\mathbf{M}\ddot{\mathbf{q}}(t) + \mathbf{N}(t) - \mathbf{f}(t) = \mathbf{0} \quad \forall t \in \mathbb{I} \tag{6.81}$$

The asymptotic series expansions of $\mathbf{N}$, $\boldsymbol{\nu}$, and $\dot{\boldsymbol{\nu}}$, about time $t = t_{n+\alpha} := (1-\alpha)t_n + \alpha t_{n+1} \in [t_n, t_{n+1}]$ for $\alpha \in [0,1]$, yields

$$\begin{aligned}\mathbf{N}(t_{n+\alpha}) &\cong \mathbf{N}(t_n) + \Theta_1 \dot{\mathbf{N}}(t_n)[t_{n+\alpha} - t_n] + \Theta_2 \ddot{\mathbf{N}}(t_n)[t_{n+\alpha} - t_n]^2 \\ &\quad + \Theta_3 \frac{\ddot{\mathbf{N}}(t_{n+1}) - \ddot{\mathbf{N}}(t_n)}{\Delta t}[t_{n+\alpha} - t_n]^3 =: \hat{\mathbf{N}} \qquad (6.82)\\ \ddot{\mathbf{q}}(t_{n+\alpha}) &\cong \ddot{\mathbf{q}}(t_n) + \Lambda_6 \frac{\ddot{\mathbf{q}}(t_{n+1}) - \ddot{\mathbf{q}}(t_n)}{\Delta t} =: \hat{\mathbf{a}} \qquad (6.83)\end{aligned}$$

respectively with new scalar algorithmic parameters $\Theta_i \in \mathbb{R}$ $(i = 1, 2, 3)$, and the linear approximation of the time-dependent external force within $[t_n, t_{n+1}]$,

$$\mathbf{f}(t_{n+\alpha}) \cong \mathbf{f}(t_n) + \frac{\mathbf{f}(t_{n+1}) - \mathbf{f}(t_n)}{\Delta t}[t_{n+\alpha} - t_n] =: \hat{\mathbf{f}} \qquad (6.84)$$

Substituting Eq. (6.82), Eq. (6.83), and Eq. (6.84) into (6.81), the residual, $\mathbf{r}_{\mathrm{I}} \neq \mathbf{0}$, is defined as

$$\mathbf{r}_{\mathrm{I}} := \mathbf{M}\hat{\mathbf{a}} + \hat{\mathbf{N}} - \hat{\mathbf{f}} \qquad (6.85)$$

Let $\tau := t_{n+\alpha} - t_n = \alpha \Delta t$. By directly employing the ***classical (traditional) time weighted residual methodology***, we have

$$\frac{\int_0^{\Delta t} W \mathbf{r} d\tau}{\int_0^{\Delta t} W d\tau} = \frac{\int_0^{\Delta t} W \left[\mathbf{M}\hat{\mathbf{a}} + \hat{\mathbf{N}} - \hat{\mathbf{f}}\right] d\tau}{\int_0^{\Delta t} W d\tau} \cong \mathbf{0} \qquad (6.86)$$

or

$$\tilde{\mathbf{r}}_{\mathrm{I}} = \mathbf{M}\tilde{\mathbf{a}} + \tilde{\mathbf{N}} - \tilde{\mathbf{f}} \cong \mathbf{0} \qquad (6.87)$$

where

$$\tilde{\mathbf{a}} := \frac{\int_0^{\Delta t} W \hat{\mathbf{a}} d\tau}{\int_0^{\Delta t} W d\tau} \quad \text{and} \quad \tilde{\mathbf{N}} := \frac{\int_0^{\Delta t} W \hat{\mathbf{N}} d\tau}{\int_0^{\Delta t} W d\tau} \qquad (6.88)$$

The degenerated scalar polynomial weighting time field for the single-field form is given as

$$W(\Gamma) \cong w_0 + w_1 \Gamma + w_2 \Gamma^2 + w_3 \Gamma^3 \quad (w_0 = 1); \qquad (6.89)$$

The replacements

$$\begin{aligned}
\frac{\int_0^{\Delta t} \frac{\tau}{\Delta t} W d\tau}{\int_0^{\Delta t} W d\tau} &\longrightarrow W_1 \\
\frac{\int_0^{\Delta t} \left(\frac{\tau}{\Delta t}\right)^2 W d\tau}{\int_0^{\Delta t} W d\tau} &\longrightarrow W_2 \\
\frac{\int_0^{\Delta t} \left(\frac{\tau}{\Delta t}\right)^3 W d\tau}{\int_0^{\Delta t} W d\tau} &\longrightarrow W_3
\end{aligned} \tag{6.90}$$

yields,

$$\tilde{\mathbf{N}} = \mathbf{N}_n + \Delta t W_1 \Theta_1 \dot{\mathbf{N}}_n + \Delta t^2 W_2 \Theta_2 \ddot{\mathbf{N}}_n + \Delta t^2 W_3 \Theta_3 (\ddot{\mathbf{N}}_{n+1} - \ddot{\mathbf{N}}_n) \tag{6.91}$$

$$\tilde{\mathbf{a}} = (1 - W_1 \Lambda_6) \ddot{\mathbf{q}}_n + W_1 \Lambda_6 \ddot{\mathbf{q}}_{n+1} \tag{6.92}$$

$$\tilde{\mathbf{f}} = \mathbf{f}_n + W_1 (\mathbf{f}_{n+1} - \mathbf{f}_n) \tag{6.93}$$

The corresponding update for $\mathbf{N}$ is designed with a new scalar algorithmic parameters $\theta_i \in \mathbb{R}$ $(i = 1, 2, 3)$ as

$$\mathbf{N}_{n+1} = \mathbf{N}_n + \Delta t \theta_1 \dot{\mathbf{N}}_n + \Delta t^2 \theta_2 \ddot{\mathbf{N}}_{n+1} + \Delta t^2 \theta_3 (\ddot{\mathbf{N}}_{n+1} - \ddot{\mathbf{N}}_n) \tag{6.94}$$

Substituting Eq. (6.94) into Eq. (6.91), we can express the algorithmic $\mathbf{N}$ vector as

$$\begin{aligned}
\tilde{\mathbf{N}} = \mathbf{N}_n &+ \frac{W_3 \Lambda_3}{\lambda_3} (\mathbf{N}_{n+1} - \mathbf{N}_n) \\
&+ \Delta t \left(W_1 \Lambda_1 - \frac{\lambda_1}{\lambda_3} W_3 \Lambda_3 \right) \dot{\mathbf{N}}_n + \Delta t^2 \left(W_2 \Lambda_2 - \frac{\lambda_2}{\lambda_3} W_3 \Lambda_3 \right) \ddot{\mathbf{N}}_n
\end{aligned} \tag{6.95}$$

Note that $\mathbf{N}_{n+1} = \mathbf{N}(\mathbf{q}_{n+1}, \boldsymbol{\nu}_{n+1}) = \mathbf{f}^{\text{int}}(\mathbf{q}_{n+1}) - \mathbf{f}^{\text{c}}(\mathbf{q}_{n+1}) - \mathbf{f}^{\text{diss}}(\boldsymbol{\nu}_{n+1}) \approx \mathbf{N}(t_{n+1})$; hence, take $\mathbf{q}_{n+1} \approx \mathbf{q}(t_n + 1)$ and $\dot{\mathbf{q}}_{n+1} \approx \dot{\mathbf{q}}(t_n + 1)$ from

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \lambda_1 \dot{\mathbf{q}}_n \Delta t + \lambda_2 \ddot{\mathbf{q}}_n \Delta t^2 + \lambda_3 (\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n) \Delta t^2 \tag{6.96}$$

$$\dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_n + \lambda_4 \ddot{\mathbf{q}}_n \Delta t + \lambda_5 (\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n) \Delta t \tag{6.97}$$

If the balance equation is given as Eq. (6.79), we have the velocity vector explicitly. In this case, we have

$$\tilde{\mathbf{r}}_{\mathrm{I}} = \mathbf{M}\tilde{\mathbf{a}} + \mathbf{C}\tilde{\mathbf{v}} + \tilde{\mathbf{f}}^{\mathrm{pot}} - \tilde{\mathbf{f}} \cong \mathbf{0} \tag{6.98}$$

where the algorithmic velocity vector is given by

$$\tilde{\mathbf{v}} = \dot{\mathbf{q}}_n + W_1\Lambda_4\ddot{\mathbf{q}}_n\Delta t + W_2\Lambda_5(\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n)\Delta t \tag{6.99}$$

from $\tilde{\mathbf{v}} := (\int_0^{\Delta t} W\hat{\mathbf{v}}d\tau)/(\int_0^{\Delta t} Wd\tau)$, where

$$\hat{\mathbf{v}} := \dot{\mathbf{q}}(t_{n+\alpha}) \cong \dot{\mathbf{q}}(t_n) + \Lambda_4\ddot{\mathbf{q}}(t_n)\tau + \Lambda_5\frac{\ddot{\mathbf{q}}(t_{n+1}) - \ddot{\mathbf{q}}(t_n)}{\Delta t}\tau^2 \tag{6.100}$$

and the algorithmic total potential force vector is

$$\begin{aligned}\tilde{\mathbf{f}}^{\mathrm{pot}} &= \mathbf{f}_n^{\mathrm{pot}} + \frac{W_3\Lambda_3}{\lambda_3}(\mathbf{f}_{n+1}^{\mathrm{pot}} - \mathbf{f}_n^{\mathrm{pot}}) \\ &+ \Delta t\left(W_1\Lambda_1 - \frac{\lambda_1}{\lambda_3}W_3\Lambda_3\right)\dot{\mathbf{f}}_n^{\mathrm{pot}} + \Delta t^2\left(W_2\Lambda_2 - \frac{\lambda_2}{\lambda_3}W_3\Lambda_3\right)\ddot{\mathbf{f}}_n^{\mathrm{pot}}\end{aligned} \tag{6.101}$$

We directly employ the algorithmic parameters from Algorithm 2 and Algorithm 3 as their extensions to nonlinear dynamical systems by *Option I*.

Algorithm 15

U0 Family-Based I-GSSSS Family of Algorithms for Nonlinear Dynamical Systems (Single-Field Form): Option I

Integrator:

$$\mathbf{M}\tilde{\mathbf{a}} + \tilde{\mathbf{N}} = \tilde{\mathbf{f}}$$

where

$$\begin{aligned}\tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1\Lambda_6\Delta\mathbf{a} \\ \tilde{\mathbf{N}} &= \mathbf{N}_n + \frac{W_3\Lambda_3}{\lambda_3}(\mathbf{N}_{n+1} - \mathbf{N}_n) \\ &\quad + \Delta t\left(W_1\Lambda_1 - \frac{\lambda_1}{\lambda_3}W_3\Lambda_3\right)\dot{\mathbf{N}}_n + \Delta t^2\left(W_2\Lambda_2 - \frac{\lambda_2}{\lambda_3}W_3\Lambda_3\right)\ddot{\mathbf{N}}_n \\ \tilde{\mathbf{f}} &= (1 - W_1)\mathbf{f}_n + W_1\mathbf{f}_{n+1} \;\; or \;\; \mathbf{f}(t_{n+W_1})\end{aligned}$$

Updates:

$$\begin{aligned}
\mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_1 \dot{\mathbf{q}}_n \Delta t + \lambda_2 \ddot{\mathbf{q}}_n \Delta t^2 + \lambda_3 \Delta \mathbf{a} \Delta t^2 \\
\dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4 \ddot{\mathbf{q}}_n \Delta t + \lambda_5 \Delta \mathbf{a} \Delta t \\
\ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_n + \Delta \mathbf{a}
\end{aligned}$$

Initial conditions:

$$\begin{aligned}
\mathbf{q}(t_0) &= \mathbf{q}_0 \\
\dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0
\end{aligned}$$

Algorithmic parameters:

$$\begin{aligned}
W_1\Lambda_1 &= \frac{1}{1+\rho_\infty^{\mathrm{s}}} \quad , \quad \lambda_1 = 1 \\
W_2\Lambda_2 &= \frac{1}{2(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_2 = \frac{1}{2} \\
W_3\Lambda_3 &= \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_3 = \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \\
W_1\Lambda_4 &= \frac{1}{1+\rho_\infty^{\mathrm{s}}} \quad , \quad \lambda_4 = 1 \\
W_2\Lambda_5 &= \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_5 = \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \\
W_1\Lambda_6 &= \frac{2+\rho_\infty^{\min}+\rho_\infty^{\max}+\rho_\infty^{\mathrm{s}}-\rho_\infty^{\min}\rho_\infty^{\max}\rho_\infty^{\mathrm{s}}}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})}
\end{aligned}$$

The weighted time field is suggested to be

$$W = 1 - \frac{15(1-2\rho_\infty^{\mathrm{s}})}{1-4\rho_\infty^{\mathrm{s}}}\frac{\tau}{\Delta t} + \frac{15(3-4\rho_\infty^{\mathrm{s}})}{1-4\rho_\infty^{\mathrm{s}}}\left(\frac{\tau}{\Delta t}\right)^2 - \frac{35(1-\rho_\infty^{\mathrm{s}})}{1-4\rho_\infty^{\mathrm{s}}}\left(\frac{\tau}{\Delta t}\right)^3$$

Algorithm 16

V0 Family-Based I-GSSSS Family of Algorithms for Nonlinear Dynamical Systems (Single-Field Form): Option I

Integrator:

$$\mathbf{M}\tilde{\mathbf{a}} + \tilde{\mathbf{N}} = \tilde{\mathbf{f}}$$

where

$$\begin{aligned}
\tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1\Lambda_6\Delta\mathbf{a} \\
\tilde{\mathbf{N}} &= \mathbf{N}_n + \frac{W_3\Lambda_3}{\lambda_3}(\mathbf{N}_{n+1} - \mathbf{N}_n) \\
&\quad + \Delta t\left(W_1\Lambda_1 - \frac{\lambda_1}{\lambda_3}W_3\Lambda_3\right)\dot{\mathbf{N}}_n + \Delta t^2\left(W_2\Lambda_2 - \frac{\lambda_2}{\lambda_3}W_3\Lambda_3\right)\ddot{\mathbf{N}}_n \\
\tilde{\mathbf{f}} &= (1 - W_1)\mathbf{f}_n + W_1\mathbf{f}_{n+1} \;\; \textit{or} \;\; \mathbf{f}(t_{n+W_1})
\end{aligned}$$

Updates:

$$\begin{aligned}
\mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + \lambda_3\Delta\mathbf{a}\Delta t^2 \\
\dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5\Delta\mathbf{a}\Delta t \\
\ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_n + \Delta\mathbf{a}
\end{aligned}$$

Initial conditions:

$$\begin{aligned}
\mathbf{q}(t_0) &= \mathbf{q}_0 \\
\dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0
\end{aligned}$$

Algorithmic parameters:

$$\begin{aligned}
W_1\Lambda_1 &= \frac{3 + \rho_\infty^{\min} + \rho_\infty^{\max} - \rho_\infty^{\min}\rho_\infty^{\max}}{2(1 + \rho_\infty^{\min})(1 + \rho_\infty^{\max})} \quad , \quad \lambda_1 = 1 \\
W_2\Lambda_2 &= \frac{1}{(1 + \rho_\infty^{\min})(1 + \rho_\infty^{\max})} \quad , \quad \lambda_2 = \frac{1}{2} \\
W_3\Lambda_3 &= \frac{1}{(1 + \rho_\infty^{\min})(1 + \rho_\infty^{\max})(1 + \rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_3 = \frac{1}{2(1 + \rho_\infty^{\mathrm{s}})} \\
W_1\Lambda_4 &= \frac{3 + \rho_\infty^{\min} + \rho_\infty^{\max} - \rho_\infty^{\min}\rho_\infty^{\max}}{2(1 + \rho_\infty^{\min})(1 + \rho_\infty^{\max})} \quad , \quad \lambda_4 = 1 \\
W_2\Lambda_5 &= \frac{2}{(1 + \rho_\infty^{\min})(1 + \rho_\infty^{\max})(1 + \rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_5 = \frac{1}{1 + \rho_\infty^{\mathrm{s}}} \\
W_1\Lambda_6 &= \frac{2 + \rho_\infty^{\min} + \rho_\infty^{\max} + \rho_\infty^{\mathrm{s}} - \rho_\infty^{\min}\rho_\infty^{\max}\rho_\infty^{\mathrm{s}}}{(1 + \rho_\infty^{\min})(1 + \rho_\infty^{\max})(1 + \rho_\infty^{\mathrm{s}})}
\end{aligned}$$

The weighted time field is suggested to be

$$\begin{aligned} W =& 1 - \frac{30(3-4\rho_\infty^{\min}-4\rho_\infty^{\max}+6\rho_\infty^{\min}\rho_\infty^{\max})}{9-11\rho_\infty^{\min}-11\rho_\infty^{\max}+19\rho_\infty^{\min}\rho_\infty^{\max}}\frac{\tau}{\Delta t} \\ &+ \frac{15(25-37\rho_\infty^{\min}-37\rho_\infty^{\max}+53\rho_\infty^{\min}\rho_\infty^{\max})}{2(9-11\rho_\infty^{\min}-11\rho_\infty^{\max}+19\rho_\infty^{\min}\rho_\infty^{\max})}\left(\frac{\tau}{\Delta t}\right)^2 \\ &- \frac{35(3-5\rho_\infty^{\min}-5\rho_\infty^{\max}+7\rho_\infty^{\min}\rho_\infty^{\max})}{9-11\rho_\infty^{\min}-11\rho_\infty^{\max}+19\rho_\infty^{\min}\rho_\infty^{\max}}\left(\frac{\tau}{\Delta t}\right)^3 \end{aligned}$$

Remark 16 (Algorithms 15 and 16)

1. *If the balance equation is given as Eq.* (6.79)*, the integrator for Algorithms 15 and 16 may be written as the following:*

$$\mathbf{M}\tilde{\mathbf{a}} + \mathbf{C}\tilde{\mathbf{v}} + \tilde{\mathbf{f}}^{\text{pot}} = \tilde{\mathbf{f}}$$

where the algorithmic total potential force vector $\tilde{\mathbf{f}}^{\text{pot}}$ *and the algorithmic velocity vector* $\tilde{\mathbf{v}}$ *are given in Eq.* (6.101) *and Eq.* (6.99)*, respectively. Note that* $\mathbf{f}_{n+1}^{\text{pot}} = \mathbf{f}^{\text{pot}}(\mathbf{q}_{n+1}) = \mathbf{f}^{\text{int}}(\mathbf{q}_{n+1}) - \mathbf{f}^{\text{c}}(\mathbf{q}_{n+1}) = \nabla\mathscr{U}(\mathbf{q}_{n+1})$*. In linear dynamical systems,* $\mathbf{f}^{\text{pot}}(\mathbf{q}) \cong \mathbf{K}\mathbf{q}$*, the algorithmic total potential force vector yields*

$$\tilde{\mathbf{f}}^{\text{pot}} \cong \mathbf{K}\tilde{\mathbf{q}} \tag{6.102}$$

where

$$\begin{aligned} \tilde{\mathbf{q}} =& \mathbf{q}_n + \frac{W_3\Lambda_3}{\lambda_3}(\mathbf{q}_{n+1}-\mathbf{q}_n) \\ &+ \Delta t\left(W_1\Lambda_1 - \frac{\lambda_1}{\lambda_3}W_3\Lambda_3\right)\dot{\mathbf{q}}_n + \Delta t^2\left(W_2\Lambda_2 - \frac{\lambda_2}{\lambda_3}W_3\Lambda_3\right)\ddot{\mathbf{q}}_n \end{aligned} \tag{6.103}$$

Hence, Algorithms 15 and 16 recover Algorithms 2 and 3, respectively, as expected for linear dynamical systems.

2. *Note that for nonlinear dynamical systems, when selecting* $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\text{s}}) = (1,1,1)$ *in Algorithms 15 and 16, we can obtain the trapezoidal rule with the endpoint acceleration (TZR-EPA) algorithm. However, when selecting* $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\text{s}}) = (1,1,0)$ *in Algorithm 16, we can obtain the trapezoidal*

rule with the midpoint acceleration (TZR-MPA) algorithm, and it is equivalent to the trapezoidal rule in the sense of the two-field form,

$$\begin{aligned} \mathbf{M}\frac{\boldsymbol{\nu}_{n+1}-\boldsymbol{\nu}_{n+1}}{\Delta t}+\mathbf{N}_{n+1/2}&=\mathbf{f}(t_{n+1/2})\\ \frac{\mathbf{q}_{n+1}-\mathbf{q}_n}{\Delta t}&=\frac{\boldsymbol{\nu}_{n+1}+\boldsymbol{\nu}_n}{2} \end{aligned} \tag{6.104}$$

3. ***Application to the Nonlinear Elastodynamical Systems:*** *Application of Algorithms 15 and 16 for the initial-value problem in the sense of the single-field form in the nonlinear elastodynamical system yields*

$$\sum_{j=1}^{n_{\text{node}}} M_{ij}\tilde{\mathbf{a}}+\tilde{\mathbf{F}}_i^{\text{int}^h}=\int_{\mathcal{B}} N_i\rho_0\mathbf{B}(t_{n+W_1})dV+\int_{\partial\mathcal{B}_\sigma} N_i\bar{\mathbf{T}}(t_{n+W_1})dA \tag{6.105}$$

with the updates and the initial conditions, (given in Algorithms 15 and 16), where $\tilde{\mathbf{F}}^{\text{int}^h}$ *denotes the algorithmic discrete internal force vector defined by*

$$\begin{aligned} \tilde{\mathbf{F}}^{\text{int}^h}&=\mathbf{F}_n^{\text{int}^h}+\frac{W_3\Lambda_3}{\lambda_3}(\mathbf{F}_{n+1}^{\text{int}^h}-\mathbf{F}_n^{\text{int}^h})\\ &\quad+\Delta t\left(W_1\Lambda_1-\frac{\lambda_1}{\lambda_3}W_3\Lambda_3\right)\dot{\mathbf{F}}_n^{\text{int}^h}\\ &\quad+\Delta t^2\left(W_2\Lambda_2-\frac{\lambda_2}{\lambda_3}W_3\Lambda_3\right)\ddot{\mathbf{F}}_n \end{aligned} \tag{6.106}$$

Particularly note that when selecting $(\rho_\infty^{\min},\rho_\infty^{\max},\rho_\infty^{\text{s}})=(1,1,0)$ *in Algorithm 16, we can obtain the trapezoidal rule with the midpoint acceleration (TZR-MPA) algorithm, and it is equivalent to the trapezoidal rule in the sense of the two-field form given in Eq.* (6.46)*.*

4. *The framework based on Option I via the classical (traditional) time weighted residual methodology does not provide attractive time-stepping algorithms which conserve total angular momentum or total energy within a time step for any spectral conditions, although the total linear momentum within the time step is exactly conserved for any spectral conditions in the conservative system. To overcome this difficulty, a new and novel* ***normalized time weighted residual methodology*** *has been recently proposed originally in Masuri et. al. [43–45].*

Numerical Implementation Aspects of Algorithms 15 and 16: The following is the numerical implementation aspect of Algorithms 15 and 16 when the fully-discrete balance equation is given by

$$\mathbf{M}\tilde{\mathbf{a}} + \mathbf{C}\tilde{\mathbf{v}} + \tilde{\mathbf{f}}^{\text{pot}} = \tilde{\mathbf{f}} \tag{6.107}$$

`Step 0:`

Initialization. Set initial conditions at $n = 0$:

$$\mathbf{q}_n = \mathbf{q}_0 \equiv \mathbf{q}(t_0) \tag{6.108}$$

$$\dot{\mathbf{q}}_n = \dot{\mathbf{q}}_0 \equiv \dot{\mathbf{q}}(t_0) \tag{6.109}$$

$$\mathbf{f}_n = \mathbf{f}_0 \equiv \mathbf{f}(t_0) \tag{6.110}$$

and compute initial acceleration from

$$\ddot{\mathbf{q}}_n = \ddot{\mathbf{q}}_0 = \mathbf{M}^{-1}\left[-\mathbf{C}\dot{\mathbf{q}}_n - \mathbf{f}_n^{\text{pot}} + \mathbf{f}_n\right] \tag{6.111}$$

`Step 1:`

At the beginning of time step, predict the state variables as

$${}^{k}\tilde{\mathbf{a}}_{n+1} = \chi^1_{\text{Pa}}\mathbf{q}_n + \chi^2_{\text{Pa}}\dot{\mathbf{q}}_n + \chi^3_{\text{Pa}}\ddot{\mathbf{q}}_n \tag{6.112}$$

$${}^{k}\tilde{\mathbf{v}}_{n+1} = \chi^1_{\text{Pv}}\mathbf{q}_n + \chi^2_{\text{Pv}}\dot{\mathbf{q}}_n + \chi^3_{\text{Pv}}\ddot{\mathbf{q}}_n \tag{6.113}$$

$${}^{k}\tilde{\mathbf{q}}_{n+1} = \chi^1_{\text{Pd}}\mathbf{q}_n + \chi^2_{\text{Pd}}\dot{\mathbf{q}}_n + \chi^3_{\text{Pd}}\ddot{\mathbf{q}}_n \tag{6.114}$$

$${}^{k}\mathbf{q}_{n+1} = \chi^1_{\text{Pu}}\mathbf{q}_n + \chi^2_{\text{Pu}}\dot{\mathbf{q}}_n + \chi^3_{\text{Pu}}\ddot{\mathbf{q}}_n \tag{6.115}$$

$${}^{k}\tilde{\mathbf{f}}^{\text{pot}}_{n+1} = \chi^1_{\text{Pp}}\mathbf{f}^{\text{pot}}_n + \chi^2_{\text{Pp}}\dot{\mathbf{f}}^{\text{pot}}_n + \chi^3_{\text{Pp}}\ddot{\mathbf{f}}^{\text{pot}}_n + \chi^4_{\text{Pp}}({}^{k}\mathbf{f}^{\text{pot}}_{n+1} - \mathbf{f}^{\text{pot}}_n) \tag{6.116}$$

$${}^{k}\tilde{\mathbf{f}}_{n+1} = (1 - W_1)\mathbf{f}_n + W_1\mathbf{f}_{n+1} \;\text{ or }\; \mathbf{f}(t_{n+W_1}) \tag{6.117}$$

`Step 2:`

Start nonlinear iteration for time step $(n+1)$. Solve for $\Delta\boldsymbol{\delta}_{n+1} := {}^{k+1}\boldsymbol{\delta}_{n+1} - {}^{k}\boldsymbol{\delta}_{n+1}$ from

$${}^{k}\mathbf{J}_{n+1}\Delta\boldsymbol{\delta}_{n+1} = -{}^{k}\mathbf{R}_{n+1} \tag{6.118}$$

where the residual vector and the system Jacobian matrix are given by

$$
\begin{aligned}
{}^{k}\mathbf{R}_{n+1} &= \mathbf{M}\,{}^{k}\tilde{\ddot{\mathbf{q}}}_{n+1} + \mathbf{C}\,{}^{k}\tilde{\dot{\mathbf{q}}}_{n+1} + {}^{k}\tilde{\mathbf{f}^{\mathrm{pot}}}_{n+1} - {}^{k}\tilde{\mathbf{f}}_{n+1} && (6.119)\\
{}^{k}\mathbf{J}_{n+1} &= \chi_{\mathrm{Ca}}\mathbf{M} + \chi_{\mathrm{Cv}}\mathbf{C} + \chi_{\mathrm{Cu}}\frac{W_3\Lambda_3}{\lambda_3}\frac{\partial^{k}\mathbf{f}^{\mathrm{pot}}{}_{n+1}}{\partial^{k}\mathbf{q}_{n+1}} && (6.120)
\end{aligned}
$$

respectively.

`Step 3:`

Correct the variables as follows:

$$
\begin{aligned}
{}^{k+1}\tilde{\mathbf{a}}_{n+1} &= {}^{k}\tilde{\ddot{\mathbf{q}}}_{n+1} + \chi_{\mathrm{Ca}}\Delta\boldsymbol{\delta}_{n+1} && (6.121)\\
{}^{k+1}\tilde{\mathbf{v}}_{n+1} &= {}^{k}\tilde{\dot{\mathbf{q}}}_{n+1} + \chi_{\mathrm{Cv}}\Delta\boldsymbol{\delta}_{n+1} && (6.122)\\
{}^{k+1}\tilde{\mathbf{q}}_{n+1} &= {}^{k}\tilde{\mathbf{q}}_{n+1} + \chi_{\mathrm{Cd}}\Delta\delta_{n+1} && (6.123)\\
{}^{k+1}\mathbf{q}_{n+1} &= {}^{k}\mathbf{q}_{n+1} + \chi_{\mathrm{Cu}}\Delta\boldsymbol{\delta}_{n+1} && (6.124)\\
{}^{k+1}\tilde{\mathbf{f}}^{\mathrm{pot}}_{n+1} &= \chi^{1}_{\mathrm{Cp}}\mathbf{f}^{\mathrm{pot}}_{n} + \chi^{2}_{\mathrm{Cp}}\dot{\mathbf{f}}^{\mathrm{pot}}_{n} + \chi^{3}_{\mathrm{Cp}}\ddot{\mathbf{f}}^{\mathrm{pot}}_{n} + \chi^{4}_{\mathrm{Cp}}({}^{k+1}\mathbf{f}^{\mathrm{pot}}_{n+1} - \mathbf{f}^{\mathrm{pot}}_{n}) && (6.125)
\end{aligned}
$$

until the solution converges.

Convergence test: If

$$\| {}^{k}\mathbf{R}_{n+1} \| \leq \text{Tolerance} \qquad (6.126)$$

then go to `Step 1` ($n \leftarrow n+1$); else, go to `Step 4`.

then go to `Step 4`; else, go to `Step 2` ($k \leftarrow k+1$).

`Step 4:`

Update the variables as follows:

$$
\begin{aligned}
\ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_{n} + ({}^{k+1}\tilde{\mathbf{a}}_{n+1} - \ddot{\mathbf{q}}_{n})/\left(W_1\Lambda_6\right) && (6.127)\\
\dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_{n} + \lambda_4\ddot{\mathbf{q}}_{n}\Delta t + \lambda_5\left(\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_{n}\right)\Delta t && (6.128)\\
\mathbf{q}_{n+1} &= \mathbf{q}_{n} + \lambda_1\dot{\mathbf{q}}_{n}\Delta t + \lambda_2\ddot{\mathbf{q}}_{n}\Delta t^2 + \lambda_3\left(\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_{n}\right)\Delta t^2 && (6.129)
\end{aligned}
$$

`Step 5:`

Output: If simulation is not completed; go to `Step 1` ($n \leftarrow n+1$).

Start new iteration ($k \leftarrow k+1$); go to `Step 2`.

The implementation coefficients for the incremental a-,v-, and d-form representations are given in Table 6.1. All forms of representation yield identical results.

6.2.2 Option II: Symplectic-Momentum Conserving Algorithms and Designs via Normalized Time Weighted Residual Methodology

Consider the equation of motion in the single-field form represented by

$$\boxed{\begin{array}{l} \textbf{\textit{Balance Equation:}} \\ \mathbf{M}\ddot{\mathbf{q}} + \mathbf{N}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{f}(t) \\ \textbf{\textit{Initial conditions:}} \\ \mathbf{q}(t_0) = \mathbf{q}_0 \\ \dot{\mathbf{q}}(t_0) = \dot{\mathbf{q}}_0 \end{array}} \tag{6.130}$$

From the balance equation, we have

$$\mathbf{M}\ddot{\mathbf{q}}(t) + \mathbf{N}(\mathbf{q}(t), \dot{\mathbf{q}}(t)) - \mathbf{f}(t) = \mathbf{0} \quad \forall t \in \mathbb{I} \tag{6.131}$$

Following the similar procedure in Subsection 6.1.2, we apply the ***normalized time weighted residual methodology*** by approximating $\mathbf{q}(t)$, $\dot{\mathbf{q}}(t)$, $\ddot{\mathbf{q}}(t)$ by the asymptotic series expansion, and we approximate the time-dependent external

force vector $\mathbf{f}(t)$ linearly as follows:

$$\begin{aligned}
\mathbf{q}(t_{n+\alpha}) &\cong \mathbf{q}(t_n) + \Lambda_1\dot{\mathbf{q}}(t_n)[t_{n+\alpha} - t_n] + \Lambda_2\ddot{\mathbf{q}}(t_n)[t_{n+\alpha} - t_n]^2 \\
&\quad + \Lambda_3\frac{\ddot{\mathbf{q}}(t_{n+1}) - \ddot{\mathbf{q}}(t_n)}{\Delta t}[t_{n+\alpha} - t_n]^3 =: \hat{\mathbf{q}} \qquad (6.132)\\
\dot{\mathbf{q}}(t_{n+\alpha}) &\cong \dot{\mathbf{q}}(t_n) + \Lambda_4\ddot{\mathbf{q}}(t_n)[t_{n+\alpha} - t_n] \\
&\quad + \Lambda_5\frac{\ddot{\mathbf{q}}(t_{n+1}) - \ddot{\mathbf{q}}(t_n)}{\Delta t}[t_{n+\alpha} - t_n]^2 =: \hat{\mathbf{v}} \qquad (6.133)\\
\ddot{\mathbf{q}}(t_{n+\alpha}) &\cong \ddot{\mathbf{q}}(t_n) + \Lambda_6\frac{\ddot{\mathbf{q}}(t_{n+1}) - \ddot{\mathbf{q}}(t_n)}{\Delta t}[t_{n+\alpha} - t_n] =: \hat{\mathbf{q}} \qquad (6.134)
\end{aligned}$$

and

$$\mathbf{f}(t_{n+\alpha}) \cong \mathbf{f}(t_n) + \frac{\mathbf{f}(t_{n+1}) - \mathbf{f}(t_n)}{\Delta t}[t_{n+\alpha} - t_n] \qquad (6.135)$$

respectively. That is, the residual, $\mathbf{r}_{\text{II}}$, is defined as

$$\mathbf{r}_{\text{II}} := \mathbf{M}\hat{\mathbf{a}} + \mathbf{N}(\hat{\mathbf{q}}, \hat{\mathbf{v}}) - \hat{\mathbf{f}} \qquad (6.136)$$

and therefore,

$$\mathbf{M}\frac{\int_0^{\Delta t} W\hat{\mathbf{a}}d\tau}{\int_0^{\Delta t} W d\tau} + \mathbf{N}(\frac{\int_0^{\Delta t} W\hat{\mathbf{q}}d\tau}{\int_0^{\Delta t} W d\tau}, \frac{\int_0^{\Delta t} W\hat{\mathbf{v}}d\tau}{\int_0^{\Delta t} W d\tau}) - \frac{\int_0^{\Delta t} W\hat{\mathbf{f}}d\tau}{\int_0^{\Delta t} W d\tau} \cong \mathbf{0} \qquad (6.137)$$

or

$$\mathbf{M}\tilde{\mathbf{a}} + \mathbf{N}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}) - \tilde{\mathbf{f}} \cong \mathbf{0} \qquad (6.138)$$

where

$$\begin{aligned}
\tilde{\mathbf{q}} &= \mathbf{q}_n + W_1\Lambda_1\dot{\mathbf{q}}_n\Delta t + W_2\Lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + W_3\Lambda_3\Delta\mathbf{a}\Delta t^2 \qquad (6.139)\\
\tilde{\mathbf{v}} &= \dot{\mathbf{q}}_n + W_1\Lambda_4\ddot{\mathbf{q}}_n\Delta t + W_2\Lambda_5\Delta\mathbf{a}\Delta t \qquad (6.140)\\
\tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1\Lambda_6\Delta\mathbf{a} \qquad (6.141)\\
\tilde{\mathbf{f}} &= \mathbf{f}_n + W_1(\mathbf{f}_{n+1} - \mathbf{f}_n) \qquad (6.142)
\end{aligned}$$

The corresponding generalized updates are:

$$\begin{aligned}
\mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + \lambda_3\Delta\mathbf{a}\Delta t^2 \qquad (6.143)\\
\dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5\Delta\mathbf{a}\Delta t \qquad (6.144)
\end{aligned}$$

with $\Delta\mathbf{a} = \ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n$. The algorithmic parameters are directly employed from Algorithm 2 and Algorithm 3 as their extensions to nonlinear dynamical systems by *Option II*. Notice that the force vector $\mathbf{N}$ is discretized in terms of both $\tilde{\mathbf{q}}$ and $\tilde{\mathbf{v}}$. The classical or normalized time weighted residual methodology is a way of temporally discretizing the balance equation so that the algorithmic time level of the fully-discrete balance equation is designed to be $t^* = t_{n+W_1}$. If the balance equation is given as shown in Eq. (6.76), the fully-discrete one may be written as

$$\begin{aligned} \mathbf{0} &= \mathbf{M}\ddot{\mathbf{q}}(t^*) + \mathbf{f}^{\text{int}}(\mathbf{q}(t^*), \dot{\mathbf{q}}(t^*)) - \mathbf{f}^{\text{ext}}(\mathbf{q}(t^*), \dot{\mathbf{q}}(t^*), t^*) \\ &= \mathbf{M}\tilde{\mathbf{a}} + \mathbf{f}^{\text{int}}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}) - \mathbf{f}^{\text{ext}}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}, t_{n+W_1}) + \mathcal{O}(\Delta t^2) \end{aligned} \tag{6.145}$$

That is,

$$\mathbf{M}\tilde{\mathbf{a}} + \mathbf{f}^{\text{int}}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}) \cong \mathbf{f}^{\text{ext}}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}, t_{n+W_1}) \tag{6.146}$$

where $\tilde{\mathbf{q}}$, $\tilde{\mathbf{v}}$, $\tilde{\mathbf{a}}$ are given by Eq. (6.139), Eq. (6.140), and Eq. (6.141), respectively, with the updates shown in Eq. (6.143), and Eq. (6.144). In the above procedure for the normalized time weighted residual methodology by *Option II*, the unknowns are the approximations in time at

$$\begin{aligned} \mathbf{f}_n &\approx \mathbf{f}(t_n) \quad & \mathbf{f}_{n+1} &\approx \mathbf{f}(t_{n+1}) \\ \mathbf{q}_n &\approx \mathbf{q}(t_n) \quad & \mathbf{q}_{n+1} &\approx \mathbf{q}(t_{n+1}) \\ \dot{\mathbf{q}}_n &\approx \dot{\mathbf{q}}(t_n) \quad & \dot{\mathbf{q}}_{n+1} &\approx \dot{\mathbf{q}}(t_{n+1}) \\ \ddot{\mathbf{q}}_n &\approx \ddot{\mathbf{q}}(t_{n-\phi}) \quad & \ddot{\mathbf{q}}_{n+1} &\approx \ddot{\mathbf{q}}(t_{n+1-\phi}) \end{aligned} \tag{6.147}$$

where $\phi := W_1(\Lambda_6 - 1)$.

Algorithm 17

U0 Family-Based I-GSSSS Family of Algorithms for Nonlinear Dynamical Systems (Single-Field Form): Option II

Integrator:

$$\mathbf{M}\tilde{\mathbf{a}} + \mathbf{N}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}) = \tilde{\mathbf{f}}$$

where

$$\begin{aligned}
\tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1\Lambda_6\Delta\mathbf{a} \\
\tilde{\mathbf{v}} &= \dot{\mathbf{q}}_n + W_1\Lambda_4\ddot{\mathbf{q}}_n\Delta t + W_2\Lambda_5\Delta\mathbf{a}\Delta t \\
\tilde{\mathbf{q}} &= \mathbf{q}_n + W_1\Lambda_1\dot{\mathbf{q}}_n\Delta t + W_2\Lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + W_3\Lambda_3\Delta\mathbf{a}\Delta t^2 \\
\tilde{\mathbf{f}} &= (1-W_1)\mathbf{f}_n + W_1\mathbf{f}_{n+1} \;\; or \;\; \mathbf{f}(t_{n+W_1})
\end{aligned}$$

Updates:

$$\begin{aligned}
\mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + \lambda_3\Delta\mathbf{a}\Delta t^2 \\
\dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5\Delta\mathbf{a}\Delta t \\
\ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_n + \Delta\mathbf{a}
\end{aligned}$$

Initial conditions:

$$\begin{aligned}
\mathbf{q}(t_0) &= \mathbf{q}_0 \\
\dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0
\end{aligned}$$

Algorithmic parameters:

$$\begin{aligned}
W_1\Lambda_1 &= \frac{1}{1+\rho_\infty^{\mathrm{s}}} \quad , \quad \lambda_1 = 1 \\
W_2\Lambda_2 &= \frac{1}{2(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_2 = \frac{1}{2} \\
W_3\Lambda_3 &= \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_3 = \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \\
W_1\Lambda_4 &= \frac{1}{1+\rho_\infty^{\mathrm{s}}} \quad , \quad \lambda_4 = 1 \\
W_2\Lambda_5 &= \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_5 = \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \\
W_1\Lambda_6 &= \frac{2+\rho_\infty^{\min}+\rho_\infty^{\max}+\rho_\infty^{\mathrm{s}}-\rho_\infty^{\min}\rho_\infty^{\max}\rho_\infty^{\mathrm{s}}}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})}
\end{aligned}$$

The weighted time field is suggested to be

$$W = 1 - \frac{15(1-2\rho_\infty^{\mathrm{s}})}{1-4\rho_\infty^{\mathrm{s}}}\frac{\tau}{\Delta t} + \frac{15(3-4\rho_\infty^{\mathrm{s}})}{1-4\rho_\infty^{\mathrm{s}}}\left(\frac{\tau}{\Delta t}\right)^2 - \frac{35(1-\rho_\infty^{\mathrm{s}})}{1-4\rho_\infty^{\mathrm{s}}}\left(\frac{\tau}{\Delta t}\right)^3$$

Algorithm 18

V0 Family-Based I-GSSSS Family of Algorithms for Nonlinear Dynamical Systems (Single-Field Form): Option II

Integrator:

$$\mathbf{M}\tilde{\mathbf{a}} + \mathbf{N}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}) = \tilde{\mathbf{f}}$$

where

$$\begin{aligned}
\tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1\Lambda_6\Delta\mathbf{a} \\
\tilde{\mathbf{v}} &= \dot{\mathbf{q}}_n + W_1\Lambda_4\ddot{\mathbf{q}}_n\Delta t + W_2\Lambda_5\Delta\mathbf{a}\Delta t \\
\tilde{\mathbf{q}} &= \mathbf{q}_n + W_1\Lambda_1\dot{\mathbf{q}}_n\Delta t + W_2\Lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + W_3\Lambda_3\Delta\mathbf{a}\Delta t^2 \\
\tilde{\mathbf{f}} &= (1 - W_1)\mathbf{f}_n + W_1\mathbf{f}_{n+1} \;\; or \;\; \mathbf{f}(t_{n+W_1})
\end{aligned}$$

Updates:

$$\begin{aligned}
\mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + \lambda_3\Delta\mathbf{a}\Delta t^2 \\
\dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5\Delta\mathbf{a}\Delta t \\
\ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_n + \Delta\mathbf{a}
\end{aligned}$$

Initial conditions:

$$\mathbf{q}(t_0) = \mathbf{q}_0$$
$$\dot{\mathbf{q}}(t_0) = \dot{\mathbf{q}}_0$$

Algorithmic parameters:

$$\begin{aligned}
W_1\Lambda_1 &= \frac{3 + \rho_\infty^{\min} + \rho_\infty^{\max} - \rho_\infty^{\min}\rho_\infty^{\max}}{2(1 + \rho_\infty^{\min})(1 + \rho_\infty^{\max})} \quad , \quad \lambda_1 = 1 \\
W_2\Lambda_2 &= \frac{1}{(1 + \rho_\infty^{\min})(1 + \rho_\infty^{\max})} \quad , \quad \lambda_2 = \frac{1}{2} \\
W_3\Lambda_3 &= \frac{1}{(1 + \rho_\infty^{\min})(1 + \rho_\infty^{\max})(1 + \rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_3 = \frac{1}{2(1 + \rho_\infty^{\mathrm{s}})} \\
W_1\Lambda_4 &= \frac{3 + \rho_\infty^{\min} + \rho_\infty^{\max} - \rho_\infty^{\min}\rho_\infty^{\max}}{2(1 + \rho_\infty^{\min})(1 + \rho_\infty^{\max})} \quad , \quad \lambda_4 = 1 \\
W_2\Lambda_5 &= \frac{2}{(1 + \rho_\infty^{\min})(1 + \rho_\infty^{\max})(1 + \rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_5 = \frac{1}{1 + \rho_\infty^{\mathrm{s}}} \\
W_1\Lambda_6 &= \frac{2 + \rho_\infty^{\min} + \rho_\infty^{\max} + \rho_\infty^{\mathrm{s}} - \rho_\infty^{\min}\rho_\infty^{\max}\rho_\infty^{\mathrm{s}}}{(1 + \rho_\infty^{\min})(1 + \rho_\infty^{\max})(1 + \rho_\infty^{\mathrm{s}})}
\end{aligned}$$

The weighted time field is suggested to be

$$\begin{aligned} W =& 1 - \frac{30(3-4\rho_\infty^{\min}-4\rho_\infty^{\max}+6\rho_\infty^{\min}\rho_\infty^{\max})}{9-11\rho_\infty^{\min}-11\rho_\infty^{\max}+19\rho_\infty^{\min}\rho_\infty^{\max}}\frac{\tau}{\Delta t} \\ &+ \frac{15(25-37\rho_\infty^{\min}-37\rho_\infty^{\max}+53\rho_\infty^{\min}\rho_\infty^{\max})}{2(9-11\rho_\infty^{\min}-11\rho_\infty^{\max}+19\rho_\infty^{\min}\rho_\infty^{\max})}\left(\frac{\tau}{\Delta t}\right)^2 \\ &- \frac{35(3-5\rho_\infty^{\min}-5\rho_\infty^{\max}+7\rho_\infty^{\min}\rho_\infty^{\max})}{9-11\rho_\infty^{\min}-11\rho_\infty^{\max}+19\rho_\infty^{\min}\rho_\infty^{\max}}\left(\frac{\tau}{\Delta t}\right)^3 \end{aligned}$$

Remark 17 (Algorithms 17 and 18)

1. *In linear dynamical systems, $\mathbf{f}^{\text{pot}}(\mathbf{q}) \cong \mathbf{K}\mathbf{q}$, the algorithmic total potential force vector yields*

$$\tilde{\mathbf{f}}^{\text{pot}} \cong \mathbf{K}\tilde{\mathbf{q}} \tag{6.148}$$

where

$$\begin{aligned} \tilde{\mathbf{q}} =& \mathbf{q}_n + \frac{W_3\Lambda_3}{\lambda_3}(\mathbf{q}_{n+1}-\mathbf{q}_n) \\ &+ \Delta t\left(W_1\Lambda_1 - \frac{\lambda_1}{\lambda_3}W_3\Lambda_3\right)\dot{\mathbf{q}}_n + \Delta t^2\left(W_2\Lambda_2 - \frac{\lambda_2}{\lambda_3}W_3\Lambda_3\right)\ddot{\mathbf{q}}_n \end{aligned} \tag{6.149}$$

Hence, Algorithms 17 and 18 recover Algorithms 2 and 3, respectively, as expected.

2. ***Time level consistency:*** *Following the similar concepts of Theorem 4 and Theorem 5, we can define the algorithmic time level $t^* = t_{n+W_1}$ from*

$$\mathbf{0} = \mathbf{M}\ddot{\mathbf{q}}(t^*) + \mathbf{C}\dot{\mathbf{q}}(t^*) + \mathbf{f}^{\text{pot}}(t^*) - \mathbf{f}(t^*) = \mathbf{M}\tilde{\mathbf{a}} + \mathbf{C}\tilde{\mathbf{v}} + \tilde{\mathbf{f}}^{\text{pot}} - \tilde{\mathbf{f}} + \mathcal{O}(\Delta t^2) \tag{6.150}$$

with $\ddot{\mathbf{q}}_n \approx \ddot{\mathbf{q}}(t_{n-\phi})$ and $\ddot{\mathbf{q}}_{n+1} \approx \ddot{\mathbf{q}}(t_{n+1-\phi})$ for $\phi = W_1(\Lambda_6 - 1)$; $\dot{\mathbf{q}}_n \approx \dot{\mathbf{q}}(t_n)$ and $\dot{\mathbf{q}}_{n+1} \approx \dot{\mathbf{q}}(t_{n+1})$; $\mathbf{q}_n \approx \mathbf{q}(t_n)$ and $\mathbf{q}_{n+1} \approx \mathbf{q}(t_{n+1})$; and $\mathbf{f}_n \approx \mathbf{f}(t_n)$ and $\mathbf{f}_{n+1} \approx \mathbf{f}(t_{n+1})$. The second-order time accuracy is obtained for all $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\ddot{\mathbf{q}}$. In general, the equation of motion in the sinlge-field form, i.e., $\mathbf{M}\ddot{\mathbf{q}} = \mathbf{F}(\mathbf{q}, \dot{\mathbf{q}}, t)$ where $\mathbf{F}$ denotes the resulting force, is discretized in time as

$$\mathbf{M}\tilde{\mathbf{a}} = \mathbf{F}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}, t_{n+W_1}) \tag{6.151}$$

which satisfies

$$\mathbf{0} = \mathbf{M}\ddot{\mathbf{q}}(t^*) + \mathbf{F}(\mathbf{q}(t^*), \dot{\mathbf{q}}(t^*), t^*) = \mathbf{M}\tilde{\mathbf{a}} + \mathbf{F}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}, t_{n+W_1}) + \mathcal{O}(\Delta t^2) \quad (6.152)$$

where $t^* = t_{n+W_1}$.

3. *When selecting* $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}}) = (1, 1, 1)$ *in Algorithms 17 and 18, we can obtain the midpoint rule with the endpoint acceleration (MPR-EPA) algorithm. When selecting* $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}}) = (1, 1, 0)$ *in Algorithm 18, we can obtain the midpoint rule with the midpoint acceleration (MPR-MPA) algorithm, and it is equivalent to the midpoint rule given in Eq.* (6.66) *in the sense of the two-field form.*

4. ***Family of Symplectic-Momentum Conserving/Dissipative Schemes:*** *We can obtain a family of symplectic-momentum conserving schemes by selecting* $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}}) = (1, 1, \rho_\infty^{\mathrm{s}})$ *with* $\rho_\infty^{\mathrm{s}} \in [0, 1]$ *in Algorithms 17 and 18. Notice that the implicit Newmark algorithm is symplectic as shown also in [46]. Otherwise, the family of algorithms are the numerically dissipative schemes with the symplectic-momentum conserving algorithms as the basis. When the controllable numerical dissipation is turned off, we readily obtain the original symplectic-momentum conserving family of algorithms.*

5. ***Composite Schemes:*** *Using Algorithm 17 from time* t_n *to* $t_{n+1/2} := t_n + \Delta t/2$ *and three-point Euler backward method from time* $t_{n+1/2}$ *to* t_{n+1} *(two sub-step method), which can be easily generalized for* n *sub-step method, the composite time integration scheme proposed by [47,48] can be recovered by selecting* $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}}) = (1, 1, 0)$ *in Algorithm 17.*

6. ***Application to the Nonlinear Elastodynamical Systems:*** *Application of Algorithms 17 and 18 for the initial-value problem in the sense of the single-field form in the nonlinear elastodynamical system as shown in Eq.*

(3.29) *yields*

$$\sum_{j=1}^{n_{\text{node}}} M_{ij}\tilde{\mathbf{a}} + \mathbf{F}_i^{\text{int}^h}(\tilde{\mathbf{q}}) = \int_{\mathcal{B}} N_i \rho_0 \mathbf{B}(t_{n+W_1}) dV + \int_{\partial\mathcal{B}_\sigma} N_i \bar{\mathbf{T}}(t_{n+W_1}) dA \quad (6.153)$$

with the updates and the initial conditions given above, where $\mathbf{F}^{\text{int}^h}(\tilde{\mathbf{q}})$ *denotes the algorithmic discrete internal force vector evaluated by* $\tilde{\mathbf{q}}$*, i.e.,*

$$\begin{aligned} \mathbf{F}_{n\,i}^{\text{int}} &= \int_{\mathcal{B}} \tilde{\mathbf{F}}^h \cdot \bar{\mathbf{S}}^h(\tilde{\mathbf{F}}^h)\ \text{GRAD}\ N_i\, dV \\ &= \sum_{j=1}^{n_{\text{node}}} \int_{\mathcal{B}} \text{GRAD}\, N_i \cdot \bar{\mathbf{S}}^h(\tilde{\mathbf{F}}^h) \text{GRAD}\, N_j dV \tilde{\mathbf{q}}^j \end{aligned} \quad (6.154)$$

in which $\tilde{\mathbf{F}}_n^h$ *denotes the algorithmic discrete deformation gradient tensor field defined by*

$$\tilde{\mathbf{F}}^h := \mathbf{I} + \sum_{i=1}^{n_{\text{node}}} \tilde{\mathbf{q}}^i \otimes \text{GRAD}\, N_i(\mathbf{X}) \quad (6.155)$$

Therefore, in this framework, the algorithmic discrete second Piola-Kirchhoff stress tensor field, $\tilde{\mathbf{S}}^h$*, and algorithmic discrete right Cauchy-Green strain tensor field,* $\tilde{\mathbf{C}}^h$*, are deinfed as*

$$\tilde{\mathbf{S}}^h := \hat{\mathbf{S}}^h(\tilde{\mathbf{C}}^h) = \bar{\mathbf{S}}^h(\tilde{\mathbf{F}}^h) \quad (6.156)$$

$$\tilde{\mathbf{C}}^h := \tilde{\mathbf{F}}^{h\,T}\tilde{\mathbf{F}}^h \quad (6.157)$$

respectively. When selecting $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\text{s}}) = (1, 1, 0)$ *in Algorithm 18, we can obtain the MPR-MPA algorithm which is equivalent to the midpoint rule shown in Eq.* (6.74) *in the sense of two-field form.*

Numerical Implementation Aspects of Algorithms 17 and 18:

`Step 0:`

Initialization. Set initial conditioins at $n = 0$:

$$\mathbf{q}_n = \mathbf{q}_0 \equiv \mathbf{q}(t_0) \quad (6.158)$$

$$\dot{\mathbf{q}}_n = \dot{\mathbf{q}}_0 \equiv \dot{\mathbf{q}}(t_0) \quad (6.159)$$

$$\mathbf{f}_n = \mathbf{f}_0 \equiv \mathbf{f}(t_0) \quad (6.160)$$

and compute initial acceleration from

$$\ddot{\mathbf{q}}_n = \ddot{\mathbf{q}}_0 = \mathbf{M}^{-1}\left[-\mathbf{N}(\mathbf{q}_n, \dot{\mathbf{q}}_n) + \mathbf{f}_n\right] \tag{6.161}$$

`Step 1:`

At the beggining of time step, predict the state variables as

$$\begin{aligned}
{}^{k}\tilde{\mathbf{a}}_{n+1} &= \chi^1_{\mathrm{Pa}}\mathbf{q}_n + \chi^2_{\mathrm{Pa}}\dot{\mathbf{q}}_n + \chi^3_{\mathrm{Pa}}\ddot{\mathbf{q}}_n && (6.162)\\
{}^{k}\tilde{\mathbf{v}}_{n+1} &= \chi^1_{\mathrm{Pv}}\mathbf{q}_n + \chi^2_{\mathrm{Pv}}\dot{\mathbf{q}}_n + \chi^3_{\mathrm{Pv}}\ddot{\mathbf{q}}_n && (6.163)\\
{}^{k}\tilde{\mathbf{q}}_{n+1} &= \chi^1_{\mathrm{Pd}}\mathbf{q}_n + \chi^2_{\mathrm{Pd}}\dot{\mathbf{q}}_n + \chi^3_{\mathrm{Pd}}\ddot{\mathbf{q}}_n && (6.164)\\
{}^{k}\tilde{\mathbf{f}}_{n+1} &= (1-W_1)\mathbf{f}_n + W_1\mathbf{f}_{n+1} \ \text{ or } \ \mathbf{f}(t_{n+W_1}) && (6.165)
\end{aligned}$$

`Step 2:`

Start nonlinear iteration for time step $(n+1)$. Solve for $\Delta\boldsymbol{\delta}_{n+1} := {}^{k+1}\boldsymbol{\delta}_{n+1} - {}^{k}\boldsymbol{\delta}_{n+1}$ from

$${}^{k}\mathbf{J}_{n+1}\Delta\boldsymbol{\delta}_{n+1} = -{}^{k}\mathbf{R}_{n+1} \tag{6.166}$$

where the residual vector and the system Jacobian matrix are given by

$$\begin{aligned}
{}^{k}\mathbf{R}_{n+1} &= \mathbf{M}\,{}^{k}\tilde{\mathbf{a}}_{n+1} + \mathbf{N}({}^{k}\tilde{\mathbf{q}}_{n+1}, {}^{k}\tilde{\mathbf{v}}_{n+1}) - {}^{k}\tilde{\mathbf{f}}_{n+1} && (6.167)\\
{}^{k}\mathbf{J}_{n+1} &= \chi_{\mathrm{Ca}}\mathbf{M} + \chi_{\mathrm{Cv}}D_2\mathbf{N}({}^{k}\tilde{\mathbf{q}}_{n+1}, {}^{k}\tilde{\mathbf{v}}_{n+1})\\
&\quad + \chi_{\mathrm{Cd}}D_1\mathbf{N}({}^{k}\tilde{\mathbf{q}}_{n+1}, {}^{k}\tilde{\mathbf{v}}_{n+1}) && (6.168)
\end{aligned}$$

respectively.

`Step 3:`

Correct the variables as follows:

$$\begin{aligned}
{}^{k+1}\tilde{\mathbf{a}}_{n+1} &= {}^{k}\tilde{\ddot{\mathbf{q}}}_{n+1} + \chi_{\mathrm{Ca}}\Delta\boldsymbol{\delta}_{n+1} && (6.169)\\
{}^{k+1}\tilde{\mathbf{v}}_{n+1} &= {}^{k}\tilde{\dot{\mathbf{q}}}_{n+1} + \chi_{\mathrm{Cv}}\Delta\boldsymbol{\delta}_{n+1} && (6.170)\\
{}^{k+1}\tilde{\mathbf{q}}_{n+1} &= {}^{k}\tilde{\mathbf{q}}_{n+1} + \chi_{\mathrm{Cd}}\Delta\boldsymbol{\delta}_{n+1} && (6.171)
\end{aligned}$$

until the solution converges.

Convergence test: If

$$\| {}^{k}\mathbf{R}_{n+1} \| \leq \text{Tolerance} \tag{6.172}$$

then go to `Step 4`; else, go to `Step 2` ($k \leftarrow k+1$).

`Step 4:`

Update the variables as follows:

$$\ddot{\mathbf{q}}_{n+1} = \ddot{\mathbf{q}}_n + ({}^{k+1}\tilde{\mathbf{a}}_{n+1} - \ddot{\mathbf{q}}_n)/(W_1\Lambda_6) \tag{6.173}$$

$$\dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5(\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n)\Delta t \tag{6.174}$$

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + \lambda_3(\ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n)\Delta t^2 \tag{6.175}$$

`Step 5:`

Output. If simulation is not completed; go to `Step 1` ($n \leftarrow n+1$).

Start new iteration ($k \leftarrow k+1$); go to `Step 2`.

The implementation coefficients for the incremental a-,v-, and d-form representations are given in Table 6.1.

We defer to a later chapter, the use and application of the normalized time weighted residual methodology for obtaining and designing energy-momentum algorithms and designs as Option III. Note that here also, as in Option II, we have energy-momentum based controllable numerical dissipative schemes and designs. When the controllable numerical dissipation is turned off, we readily obtain the original energy-momentum family of algorithms.

6.3 Numerical Results

In this section, various numerical results are shown with the time step size of $\Delta t = 0.01$ sec. To demonstrate and illustrate the numerical behaviors of the implicit family of algorithms presented in this chapter, we consider the following three problems: (1) Nonlinear oscillator [1], (2) (Classical) Kepler's problem, and (3) Lennard-Jones (5, 3) Potential 2-body Problem. The brief summaries of the numerical examples are shown in Appendix A, and the input information for the simulation are listed in Table 6.2. We demonstrate the implicit family of algorithms in the sense of Option I and Option II. The selected algorithms are:
Implicit GSSSS family of algorithms (Option I): Algorithms 15 and 16
Implicit GSSSS family of algorithms (Option II): Algorithms 17 and 18
Algorithms 15/17 and Algorithms 16/18 are denoted by U0V• and V0U• in the figures, respectively. For each algorithm, we arbitrarily select,
U0V0/V0U0(ρ_∞,1.0,ρ_∞)
U0V0(1.0,1.0,ρ_∞^{s})
V0U0(1.0,1.0,ρ_∞^{s})
where

$$\begin{aligned} &\rho_\infty = \rho_\infty^{\min} = \rho_\infty^{\max} \in \{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\} \\ &\rho_\infty^{\mathrm{s}} \in \{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\} \end{aligned} \tag{6.176}$$

The legends of the figures are shown in Fig. 4.5.

Nonlinear Oscillator problem [1]: The second-order time accuracies in the configuration, velocity, and acceleration in the forced-damped system, Eq. (A.31) with $C = 0.001$ and $\mathbf{f}^{\mathrm{ext}} = \bar{\mathbf{f}}\sin(t)$ where $\bar{\mathbf{f}} = (0.1, -0.5)^T$, with the selected schemes from the family of implicit GSSSS algorithms in the sense of Options I and II are shown in Fig. 6.1 and Fig. 6.2, respectively. The configuration,

velocity, and acceleration have been used to calculate the reference errors,

$$\begin{aligned} \text{error}_q &= \left| \frac{q^1 - q^1_{\text{ref}}}{q^1_{\text{ref}}} \right| \\ \text{error}_v &= \left| \frac{\dot{q}^1 - \dot{q}^1_{\text{ref}}}{\dot{q}^1_{\text{ref}}} \right| \\ \text{error}_a &= \left| \frac{\ddot{q}^1 - \ddot{q}^1_{\text{ref}}}{\ddot{q}^1_{\text{ref}}} \right| \end{aligned} \tag{6.177}$$

respectively, at time $t = 1$ sec. The reference configurations, velocities, and accelerations have been obtained with a sufficiently small time step size $\Delta t = 10^{-5}$. As can be seen from the figures, all time integration schemes are second-order time accurate in all the configurations, velocities, and accelerations. It should be noted that the second-order time accuracies of the schemes not only in the configurations and velocities, but also the accelerations have been achieved from the consistency of the time level of the discrete algorithmic balance equations at time level $t = t_{n+W_1}$.

In this problem, the total linear momentum within a time step should not be conserved; therefore, we should have $\mathbf{L}_n \neq \mathbf{L}_{n+1}$. As can be seen from Fig. 6.3 - Fig. 6.14, the linear momentum is always bounded for any schemes in the family of algorithms in Option I and Option II both in the conservative and dissipative systems as expected.

On the other hand, the conservation of the angular momentum in the sense of $\mathbf{J}_n = \mathbf{J}_{n+1}$ in the conservative system is achieved only for some special cases. That is, only the particular family of schemes with $\rho_\infty^{\min} = \rho_\infty^{\max} = \rho_\infty^{\text{s}} = 1$ in the U0-based family in Option II or $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\text{s}} \in [0, 1]$ in the V0-based family in Option II can conserve the angular momentum within the time step exactly. Fig. 6.3 and Fig. 6.6 illustrate that the U0V0/V0U0 optimal schemes in both Option I and Option II are not angular momentum conserving schemes [except (1,1,1) case in Option II]; however, they give better results in the qualitative sense by we selecting a higher ρ_∞ close to but less than unity, i.e., $\rho_\infty \leq 1$ [Note $\rho_\infty^{\min} = \rho_\infty^{\text{s}} = \rho_\infty$ and $\rho_\infty^{\max} = 1$]. In the dissipative system, all schemes show that the angular momentum is always decaying as expected.

Also, as can be seen from Fig. 6.3 - Fig. 6.14, there is no scheme which conserves the mechanical energy of the system in the sense of $\mathscr{E}_n = \mathscr{E}_{n+1}$ in the conservative system. In the dissipative system, all schemes selected show the dissipative behaviors of the mechanical energy. Since the schemes in Option II with the selections $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\mathrm{s}} \in [0, 1]$ in both U0 and V0 families are variational, the mechanical energy is always bounded. The U0V0$(1, 1, \rho_\infty^{\mathrm{s}})$ schemes in Option II show that the amplitude of the oscillation of the mechanical energy can be decreased by selecting a higher value of ρ_∞^{s} close to but less than unity, i.e., $\rho_\infty^{\mathrm{s}} \leq 1$. The U0V0/V0U0 optimal schemes show that the mechanical energy in the conservative system tends to dissipate more rapidly for lower values of ρ_∞. The initial conditions used for the simulation were:

$$\begin{aligned} \mathbf{q}(t_0) &= \mathbf{q}_0 = (0.1, 1.0)^T \\ \dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0 = (0.1, 0.1)^T \end{aligned} \tag{6.178}$$

The final time of the simulation are selected to be $T = 30$ sec.

3D Kepler's Problem: In this example problem, we applied the same selected schemes from the previous example problem. And we observed similar numerical behaviors of the algorithms in the time accuracies and the linear and angular momenta and mechanical energy conservations. Similar to the nonlinear oscillator problem presented above, the linear momentum should not be conserved in the absence of the external loadings. Fig. 6.15 and Fig. 6.16 show that the second-order time accuracies are obtained in the configuration, velocity, and acceleration in the forced-damped system, with $C = 0.01$ and the external loading $\mathbf{f}^{\mathrm{ext}} = \bar{\mathbf{f}} \sin(t)$ where $\bar{\mathbf{f}} = (0.1, 0.0, 0.0)^T$, with the selected schemes from the family of implicit GSSSS algorithms in the sense of Options I and II. The time histories of the mechanical energy, linear and angular momenta, configuration, velocity, and acceleration in the conservative system are shown in Fig. 6.17 - Fig. 6.22. The initial conditions used for the simulation were:

$$\begin{aligned} \mathbf{q}(t_0) &= \mathbf{q}_0 = (0.9/\sqrt{2}, 0.0, 0.9/\sqrt{2})^T \\ \dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0 = (0.0, -100/9, 0.0)^T \end{aligned} \tag{6.179}$$

The final time of the simulation are selected to be $T = 20$ sec. Note that the numerical results for dissipative systems are not shown for this nonlinear dynamical problem.

Lennard-Jones potential problem: Again, in this example problem, we applied the same selected schemes from the previous example problem. And we basically observed the same numerical behaviors of the algorithms in the time accuracies and the angular momenta and mechanical energy conservation. But, the total linear momentum should be conserved exactly in the sense of $\mathbf{L}_n = \mathbf{L}_{n+1}$ in this problem in the absence of the external loading. Fig. 6.25 - Fig. 6.30 show that the total linear momentum is exactly conserved in the conservative system for any choices of the spectral conditions in both Option I and Option II. Fig. 6.23 and Fig. 6.24 show that the second-order time accuracies are obtained in the configuration, velocity, and acceleration in the forced-damped system, with $C = 0.01$ and the external loading $\mathbf{f}^{\text{ext}} = \bar{\mathbf{f}} \sin(t)$ where $\bar{\mathbf{f}} = (0.1, 0.0, 0.0, 0.0, -0.5, 0.0)^T$, with the selected schemes from the family of implicit GSSSS algorithms in the sense of Options I and II. The time histories of the mechanical energy, linear and angular momenta, configuration, velocity, and acceleration in the conservative system are shown in Fig. 6.25 - Fig. 6.30. The initial conditions used for the simulation were:

$$\begin{aligned} \mathbf{q}(t_0) &= \mathbf{q}_0 = (0.0, -0.5, 0.0, 0.0, 0.5, 0.0)^T \\ \dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0 = (5.0, 1.0, 2.0, 10.0, 3.0, -1.0)^T \end{aligned} \tag{6.180}$$

The final time of the simulation are selected to be $T = 2$ sec. Note that the numerical results for dissipative systems are not shown for this nonlinear dynamical problem.

Here is the overall summary of the numerical results:

Remark 18

1. *The order of time accuracies in the configuration, velocity, and acceleration is* 2 *for any selections of the spectral conditions.*

2. *There is no energy conserving implicit scheme in the conservative system.*

3. *The angular momentum conserving schemes are obtained by selecting the following spectral conditions in the conservative system:*
   Option II: $\rho_\infty^{\min} = \rho_\infty^{\max} = \rho_\infty^{\mathrm{s}} = 1$ *in the U0-based family/*$\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ *and* $\rho_\infty^{\mathrm{s}} \in [0, 1]$ *in the V0-based family*

4. *Variational schemes are obtained by selecting the following spectral conditions in the conservative system:*
   $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ *and* $\rho_\infty^{\mathrm{s}} \in [0, 1]$ *in the U0-based family/V0-based family*

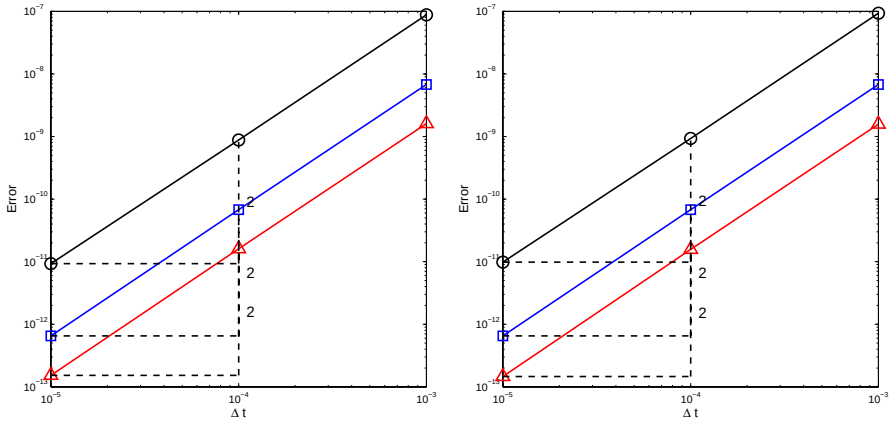
5. *In the dissipative system, all schemes show the energy and angular momentum dissipative features. When the controllable numerical dissipation is turned off, we readily recover the original symplectic-momentum conserving algorithms and designs.*

	a-form	**v-form**	**d-form**
χ^1_{Pu}	1	1	1
χ^2_{Pu}	$W_1\Lambda_1\Delta t$	$W_1\Lambda_1\Delta t$	$\left(W_1\Lambda_1 - \frac{W_3\Lambda_3\lambda_1}{\lambda_3}\right)\Delta t$
χ^3_{Pu}	$W_2\Lambda_2\Delta t^2$	$\left(W_2\Lambda_2 - W_3\Lambda_3\lambda_4\right)\Delta t^2$	$\left(W_2\Lambda_2 - \frac{W_3\Lambda_3\lambda_2}{\lambda_3}\right)\Delta t^2$
χ^1_{Pd}	1	1	1
χ^2_{Pd}	$\lambda_1\Delta t$	$\lambda_1\Delta t$	0
χ^3_{Pd}	$\lambda_2\Delta t^2$	$\left(\lambda_2 - \frac{\lambda_3\lambda_4}{\lambda_5}\right)\Delta t^2$	0
χ^1_{Pv}	0	0	0
χ^2_{Pv}	1	1	$1 - \frac{W_2\Lambda_5\lambda_1}{\lambda_3}$
χ^3_{Pv}	$W_1\Lambda_4\Delta t$	$\left(W_1\Lambda_4 - \frac{W_2\Lambda_5\lambda_4}{\lambda_5}\right)\Delta t$	$\left(W_1\Lambda_4 - \frac{W_2\Lambda_5\lambda_2}{\lambda_3}\right)\Delta t$
χ^1_{Pa}	0	0	0
χ^2_{Pa}	0	0	$-\frac{W_1\Lambda_6\lambda_1}{\lambda_3\Delta t}$
χ^3_{Pa}	1	$1 - \frac{W_1\Lambda_6\lambda_4}{\lambda_5}$	$1 - \frac{W_1\Lambda_6\lambda_2}{\lambda_3}$
χ^1_{Pp}	1	1	1
χ^2_{Pp}	$\left(W_1\Lambda_1 - \frac{W_3\Lambda_3\lambda_1}{\lambda_3}\right)\Delta t$	$\left(W_1\Lambda_1 - \frac{W_3\Lambda_3\lambda_1}{\lambda_3}\right)\Delta t$	$\left(W_1\Lambda_1 - \frac{W_3\Lambda_3\lambda_1}{\lambda_3}\right)\Delta t$
χ^3_{Pp}	$\left(W_2\Lambda_2 - \frac{W_3\Lambda_3\lambda_2}{\lambda_3}\right)\Delta t$	$\left(W_2\Lambda_2 - \frac{W_3\Lambda_3\lambda_2}{\lambda_3}\right)\Delta t$	$\left(W_2\Lambda_2 - \frac{W_3\Lambda_3\lambda_2}{\lambda_3}\right)\Delta t$
χ^4_{Pp}	$\frac{W_3\Lambda_3}{\lambda_3}$	$\frac{W_3\Lambda_3}{\lambda_3}$	$\frac{W_3\Lambda_3}{\lambda_3}$
χ_{Cu}	$\lambda_3\Delta t^2$	$\frac{\lambda_3}{\lambda_5}\Delta t$	1
χ_{Cd}	$W_3\Lambda_3\Delta t^2$	$\frac{W_3\Lambda_3}{\lambda_5}\Delta t$	$\frac{W_3\Lambda_3}{\lambda_3}$
χ_{Cv}	$W_2\Lambda_5\Delta t$	$\frac{W_2\Lambda_5}{\lambda_5}$	$\frac{W_2\Lambda_5}{\lambda_3\Delta t}$
χ_{Ca}	$W_1\Lambda_6$	$\frac{W_1\Lambda_6}{\lambda_5\Delta t}$	$\frac{W_1\Lambda_6}{\lambda_3\Delta t^2}$
χ^1_{Cp}	1	1	1
χ^2_{Cp}	$\left(W_1\Lambda_1 - \frac{W_3\Lambda_3\lambda_1}{\lambda_3}\right)\Delta t$	$\left(W_1\Lambda_1 - \frac{W_3\Lambda_3\lambda_1}{\lambda_3}\right)\Delta t$	$\left(W_1\Lambda_1 - \frac{W_3\Lambda_3\lambda_1}{\lambda_3}\right)\Delta t$
χ^3_{Cp}	$\left(W_2\Lambda_2 - \frac{W_3\Lambda_3\lambda_2}{\lambda_3}\right)\Delta t$	$\left(W_2\Lambda_2 - \frac{W_3\Lambda_3\lambda_2}{\lambda_3}\right)\Delta t$	$\left(W_2\Lambda_2 - \frac{W_3\Lambda_3\lambda_2}{\lambda_3}\right)\Delta t$
χ^4_{Cp}	$\frac{W_3\Lambda_3}{\lambda_3}$	$\frac{W_3\Lambda_3}{\lambda_3}$	$\frac{W_3\Lambda_3}{\lambda_3}$

Table 6.1: Predictor multi-corrector coefficients for the incremental a-,v-, and d-form representations

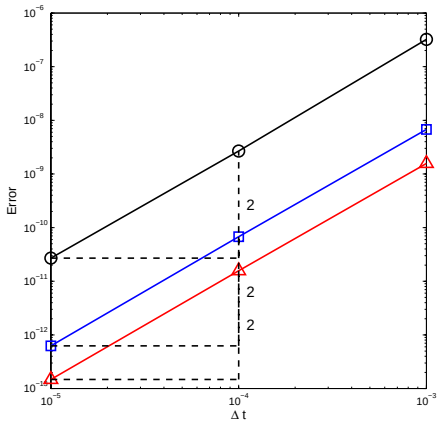
Problems	Input parameters	Initial conditions
Nonlinear oscillator	$m = 1$	$\mathbf{q}_0 = (0.50, 1.00)^T$
		$\dot{\mathbf{q}}_0 = (0.50, 0.25)^T$
Kepler's problem	$m = 1$	$\mathbf{q}_0 = (0.9/\sqrt{2}, 0, 0.9/\sqrt{2})^T$
	$k = 100$	$\dot{\mathbf{q}}_0 = (0, -100/9, 0)^T$
Lennard-Jones potential problem	$m_1 = m_2 = 1$	$\mathbf{q}_0 = (0, -0.5, 0, 0, 0.5, 0)^T$
	$\sigma = \sqrt{3/5}$, $\alpha = 100$	$\dot{\mathbf{q}}_0 = (5, 1, 2, 10, 3, -1)^T$

Table 6.2: Input parameters for the nonlinear dynamics example problems

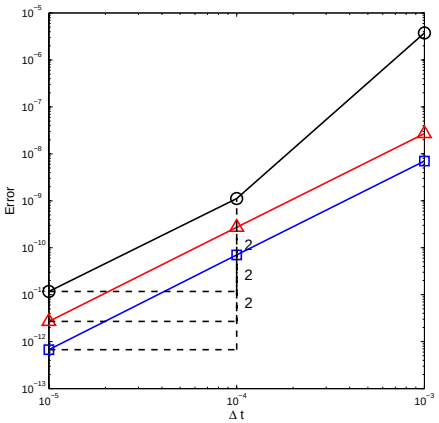


(a) U0V0/V0U0(1.0,1.0,1.0)

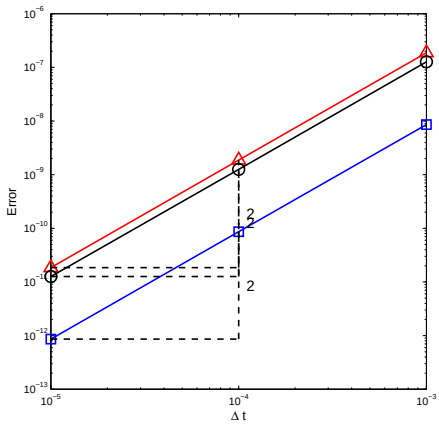
(b) U0V0(1.0,1.0,0.0)



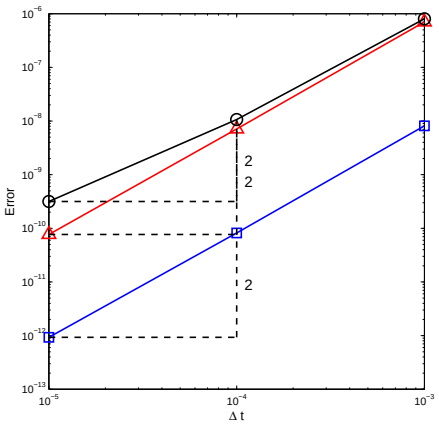
(c) V0U0(1.0,1.0,0.0)



(d) U0V0/V0U0(0.8,1.0,0.8)

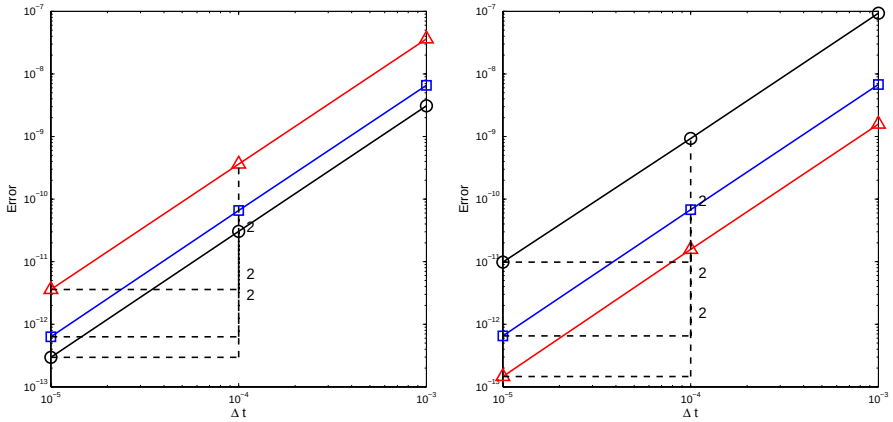


(e) U0V1(0.85,0.90,0.20)



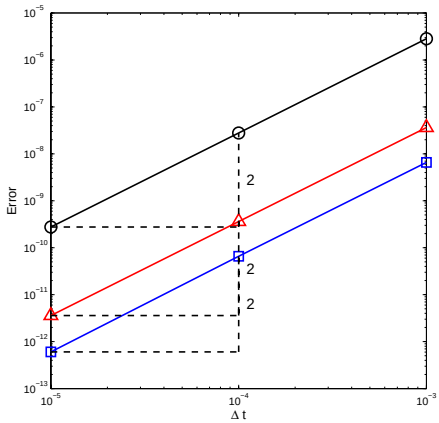
(f) V0U1(0.85,0.90,0.20)

Figure 6.1: Time accuracies in the configuration (□), velocity (△), and acceleration (○) in the forced mechanical system. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Option I)]

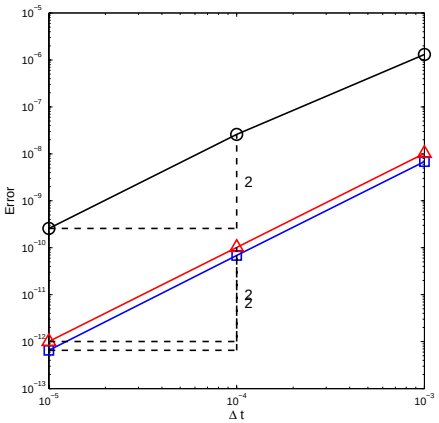


(a) U0V0/V0U0(1.0,1.0,1.0)

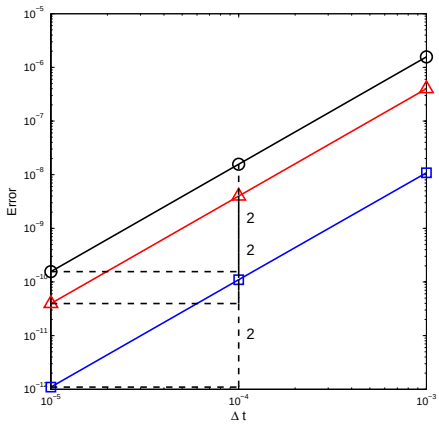
(b) U0V0(1.0,1.0,0.0)



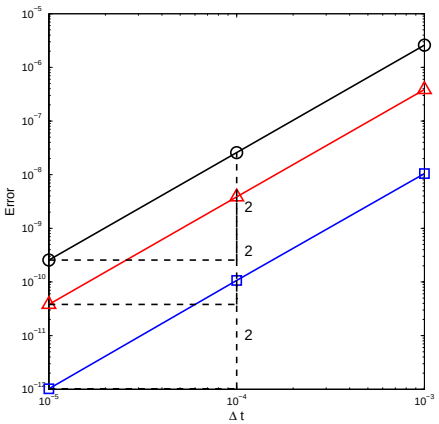
(c) V0U0(1.0,1.0,0.0)



(d) U0V0/V0U0(0.8,1.0,0.8)

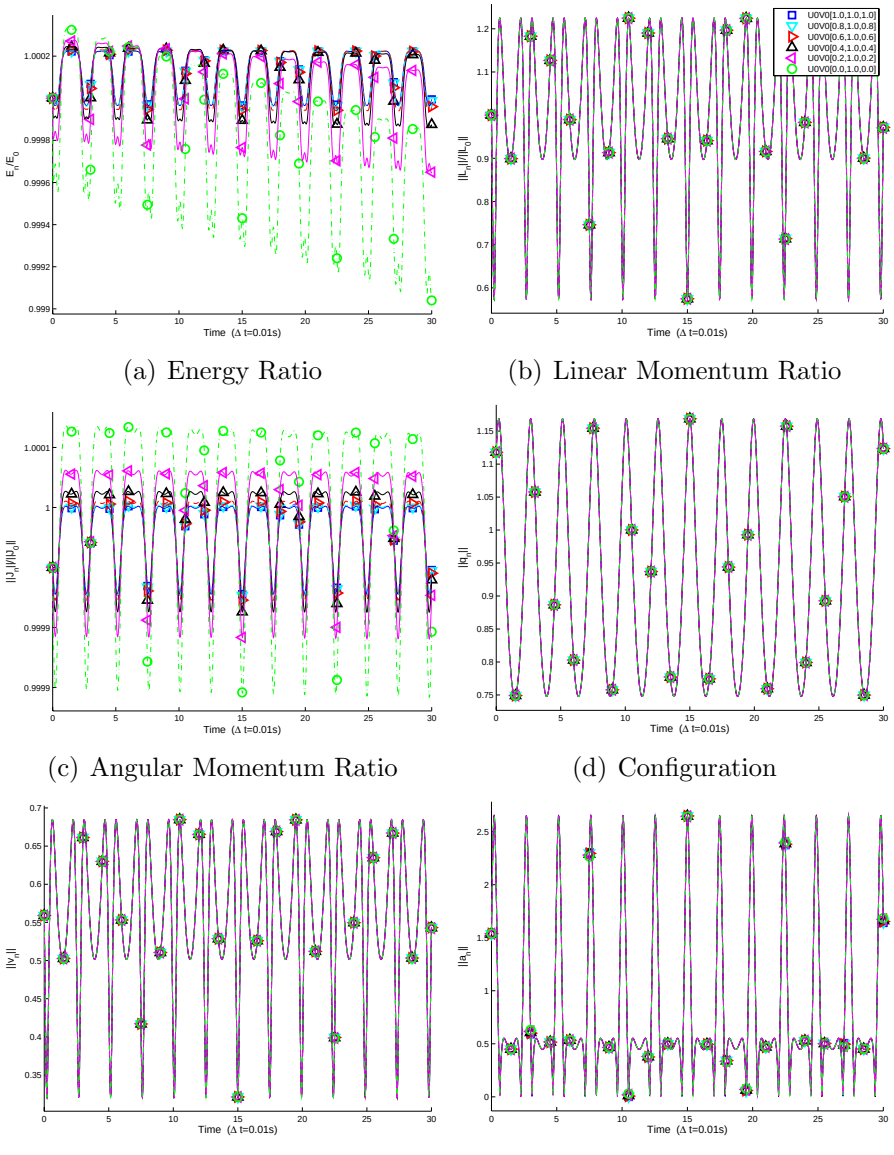


(e) U0V1(0.6,0.9,0.2)



(f) V0U1(0.6,0.9,0.2)

Figure 6.2: Time accuracies in the configuration (□), velocity (△), and acceleration (○) in the forced mechanical system [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Option II)]

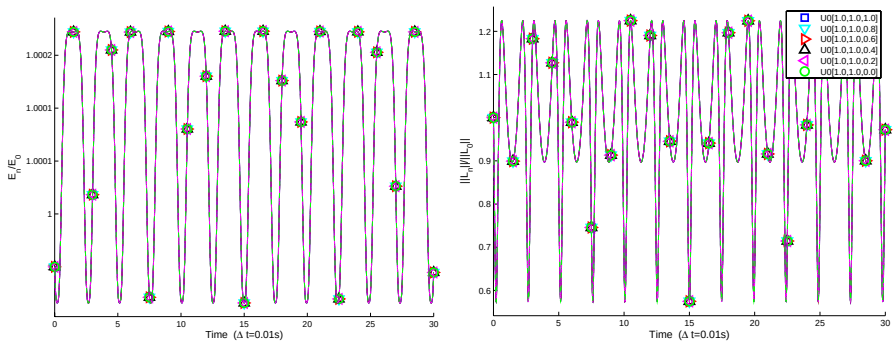


(a) Energy Ratio (b) Linear Momentum Ratio

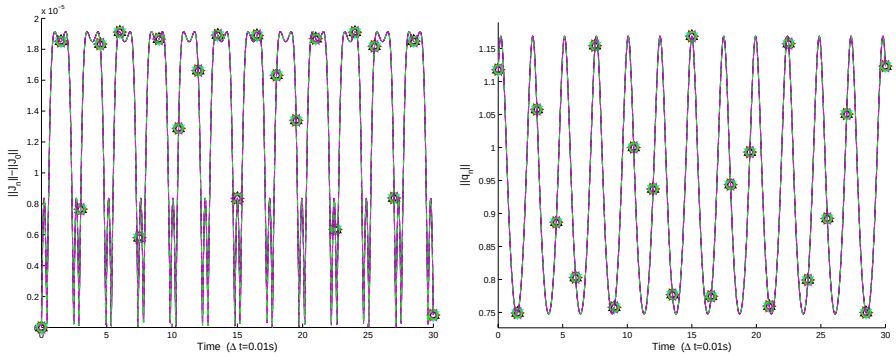
(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

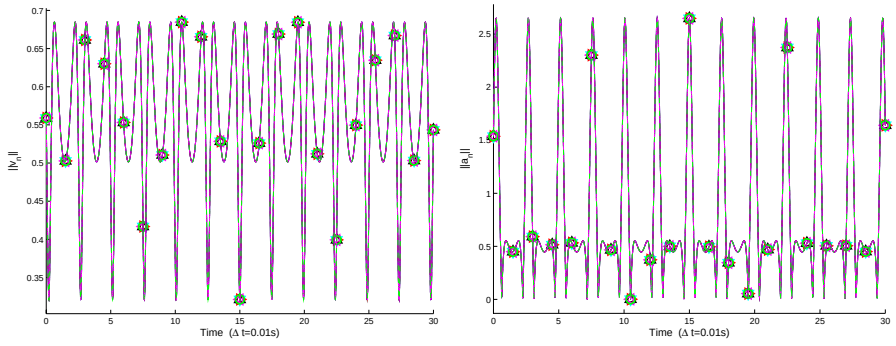
Figure 6.3: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Option I) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum Ratio

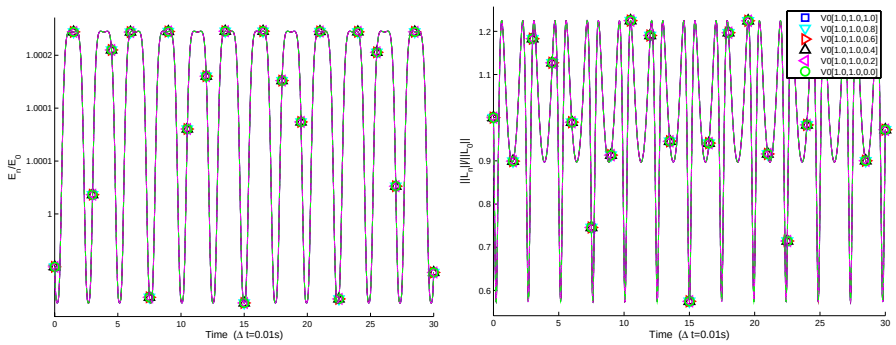


(c) Angular Momentum Difference (d) Configuration

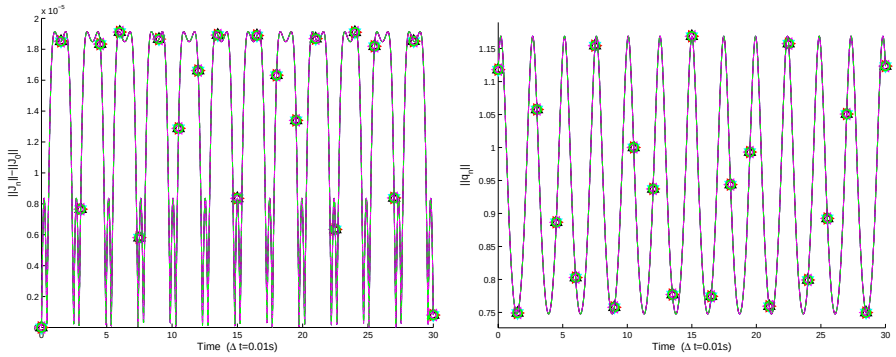


(e) Velocity Norm (f) Acceleration Norm

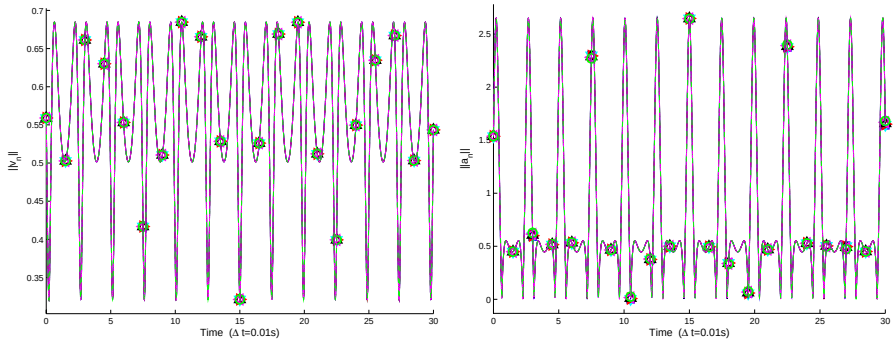
Figure 6.4: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Option I) - U0V0(1.0,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum Ratio

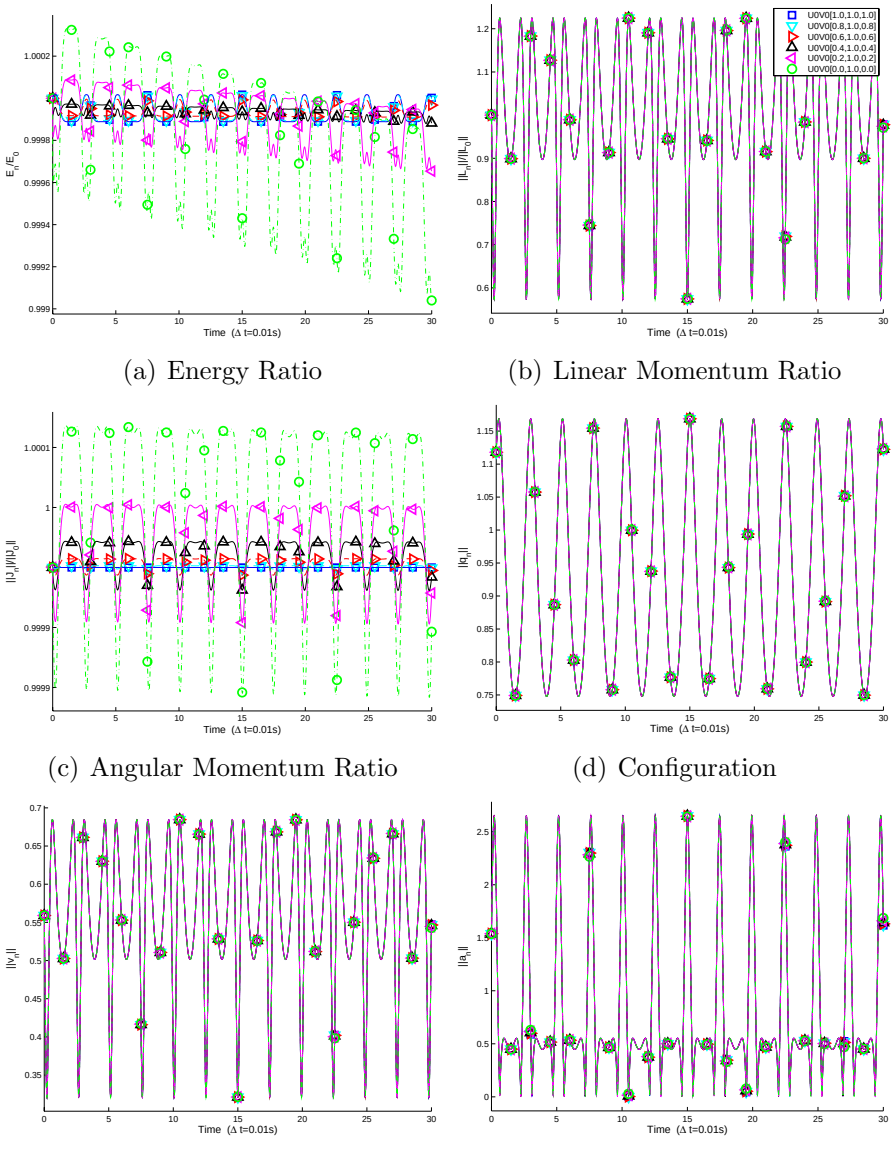


(c) Angular Momentum Difference (d) Configuration



(e) Velocity Norm (f) Acceleration Norm

Figure 6.5: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Option I) - V0U0(1.0,1.0,ρ_∞)]

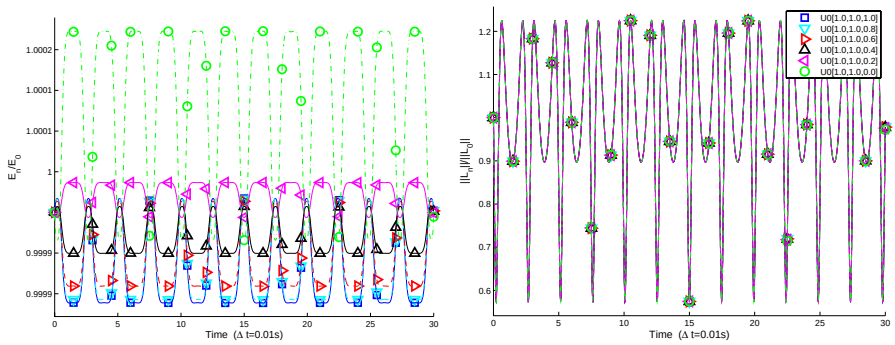


(a) Energy Ratio (b) Linear Momentum Ratio

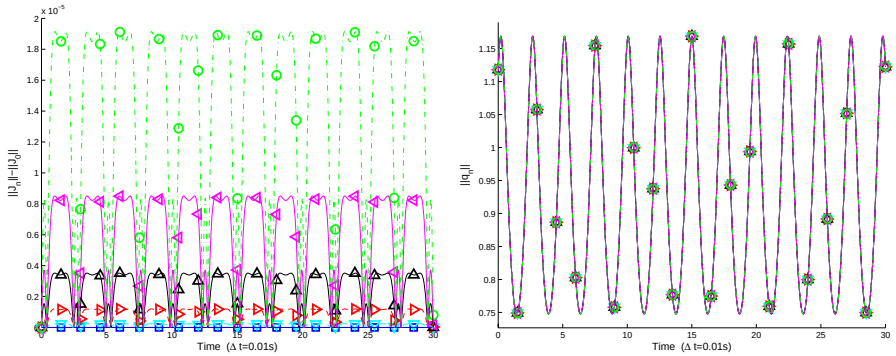
(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

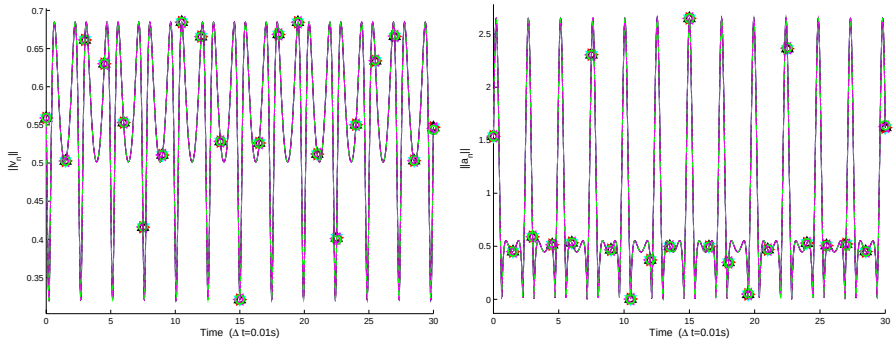
Figure 6.6: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Option II) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum Ratio

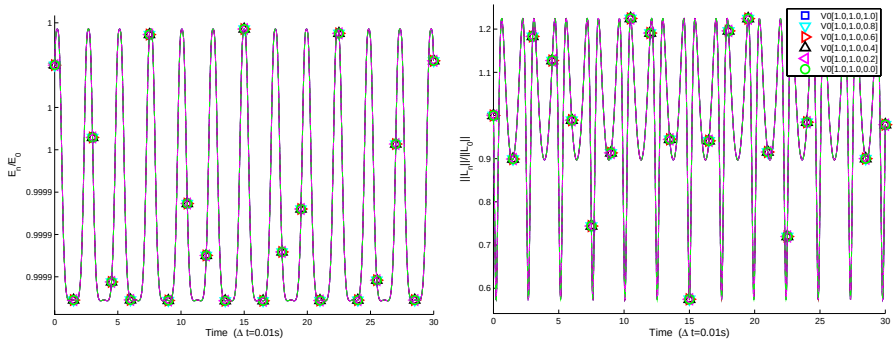


(c) Angular Momentum Difference (d) Configuration

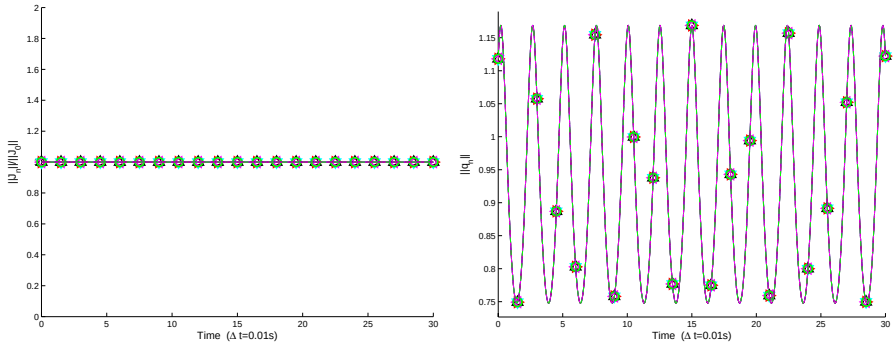


(e) Velocity Norm (f) Acceleration Norm

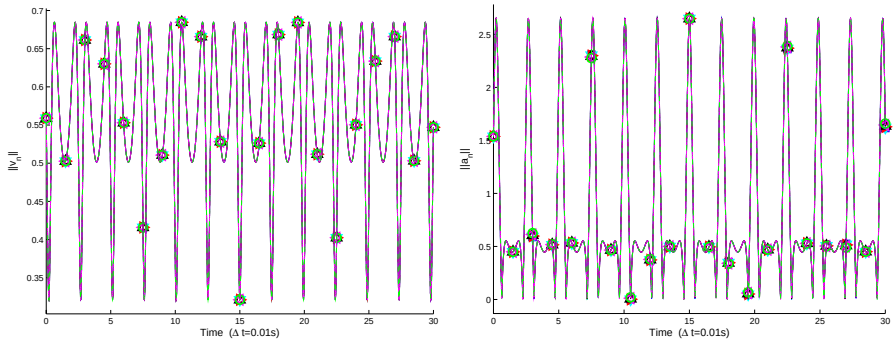
Figure 6.7: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Option II) - U0V0(1.0,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum Ratio

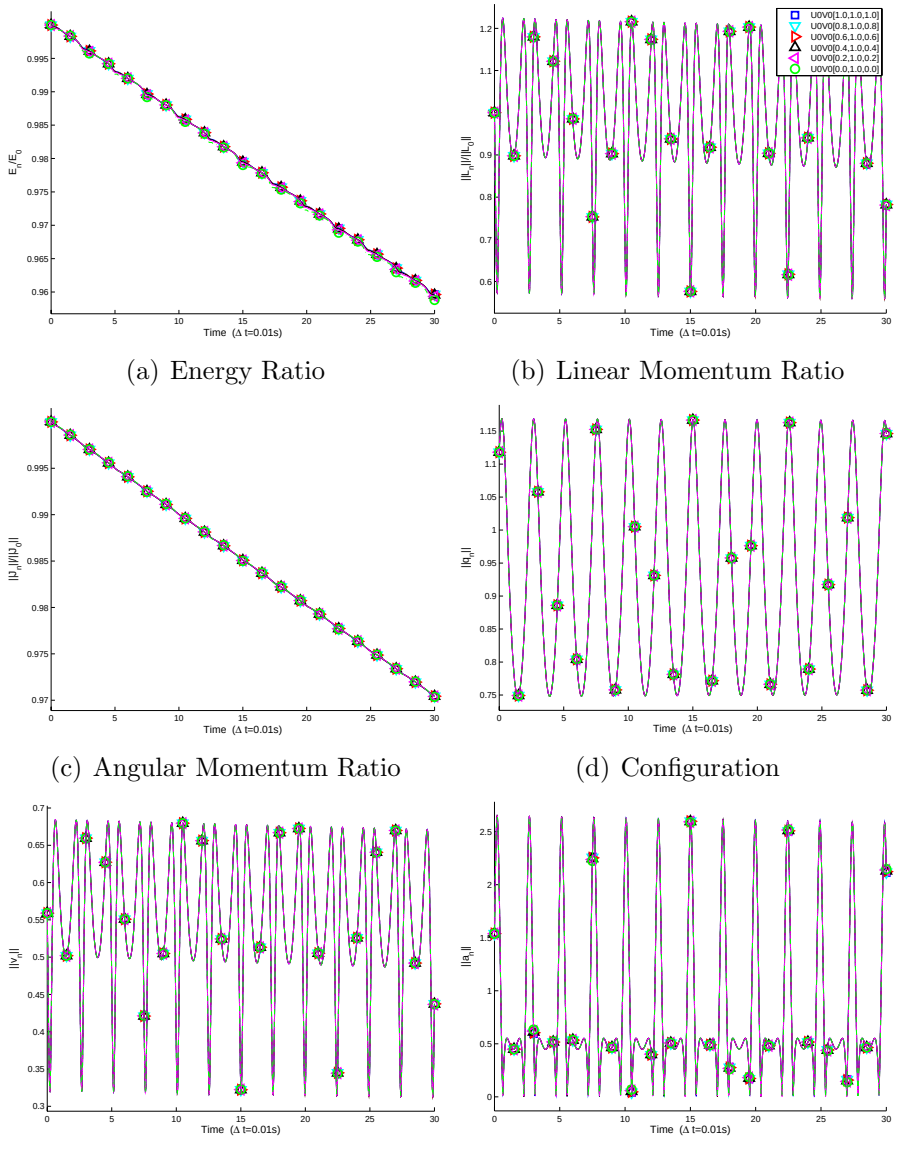


(c) Angular Momentum Ratio (d) Configuration



(e) Velocity Norm (f) Acceleration Norm

Figure 6.8: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Option II) - V0U0(1.0,1.0,ρ_∞)]

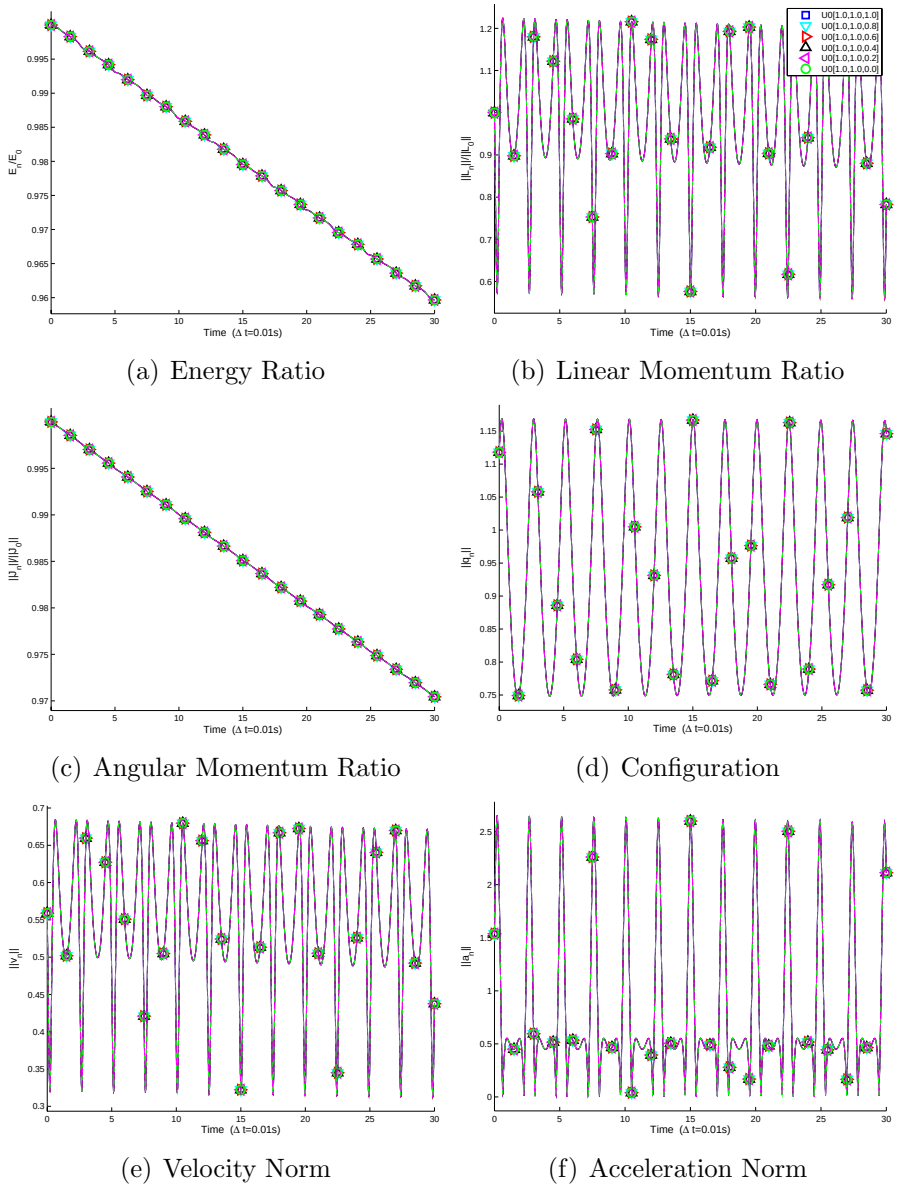


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 6.9: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Option I) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]

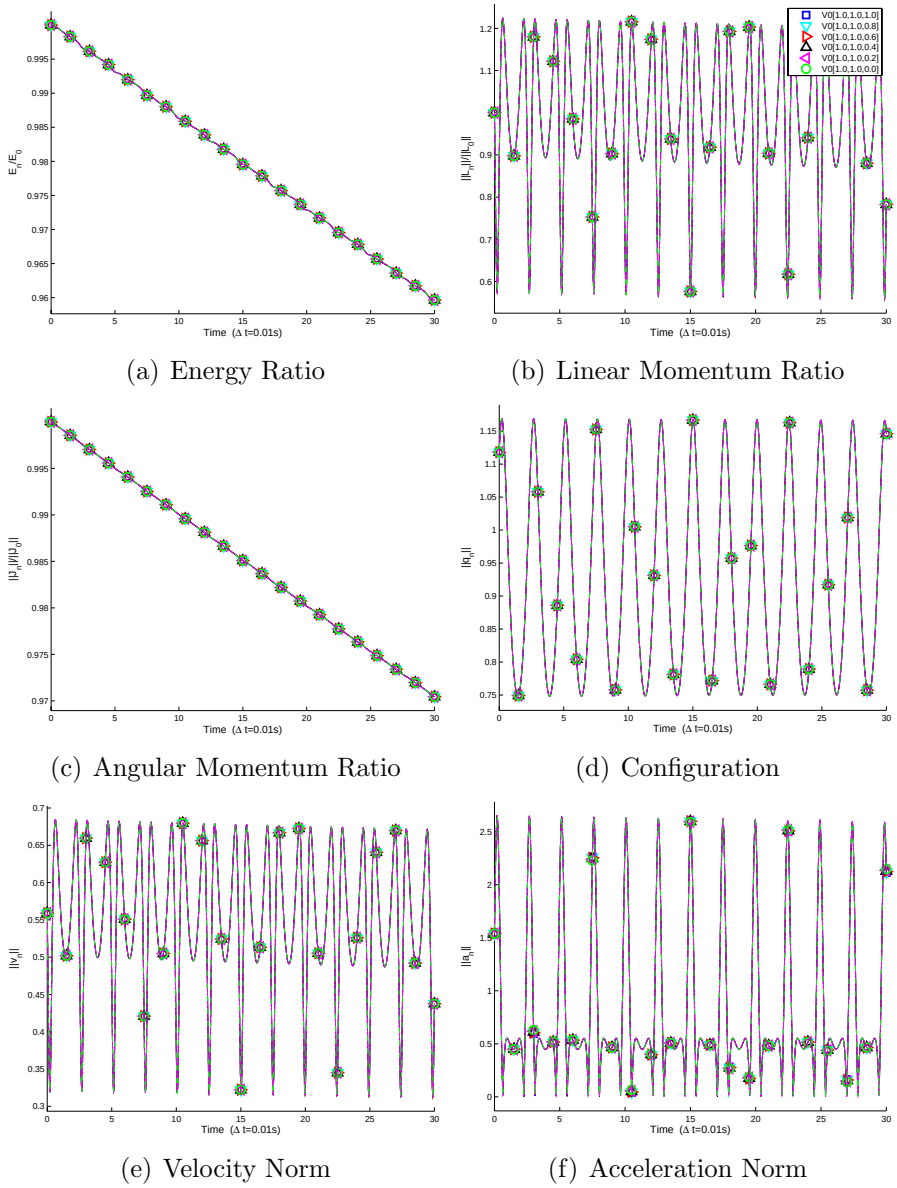


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 6.10: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Option I) - U0V0(1.0,1.0,ρ_∞)]

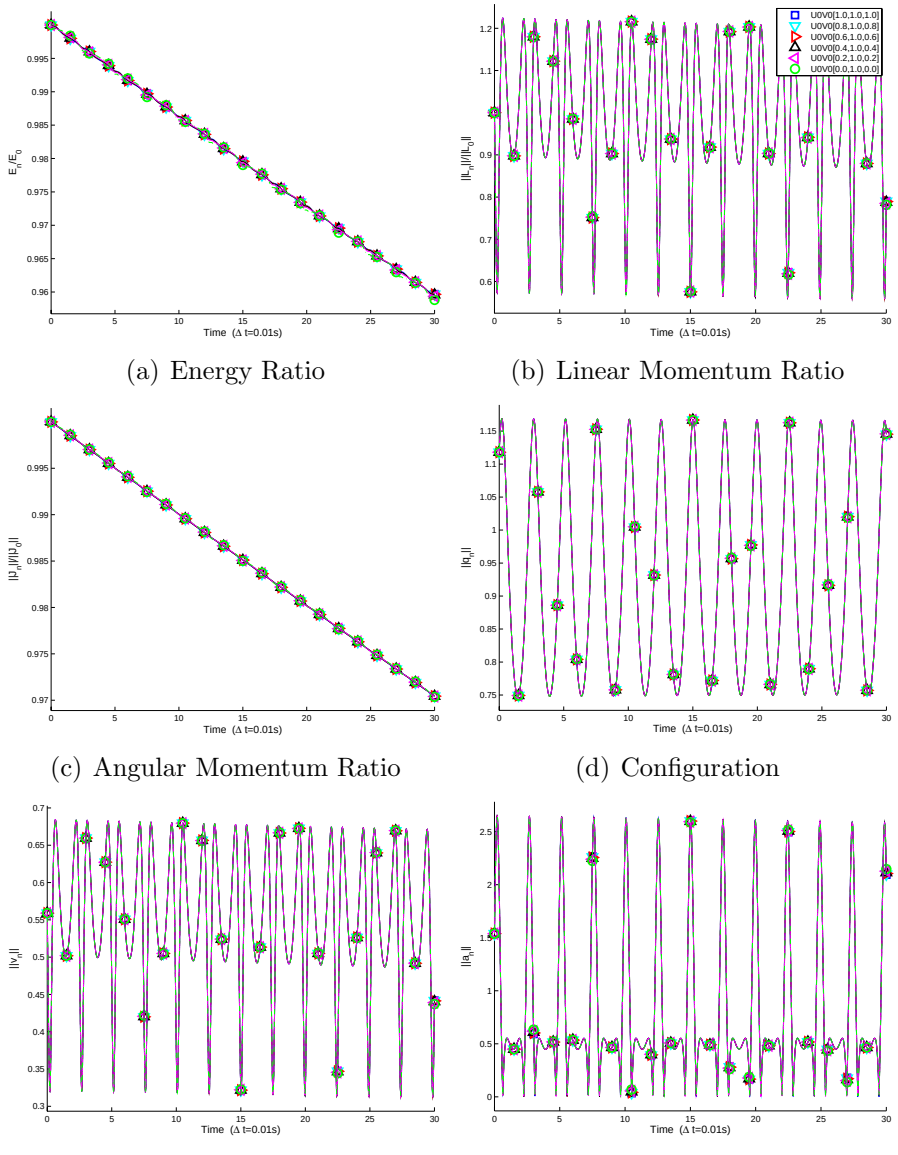


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 6.11: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Option I) - V0U0(1.0,1.0,ρ_∞)]

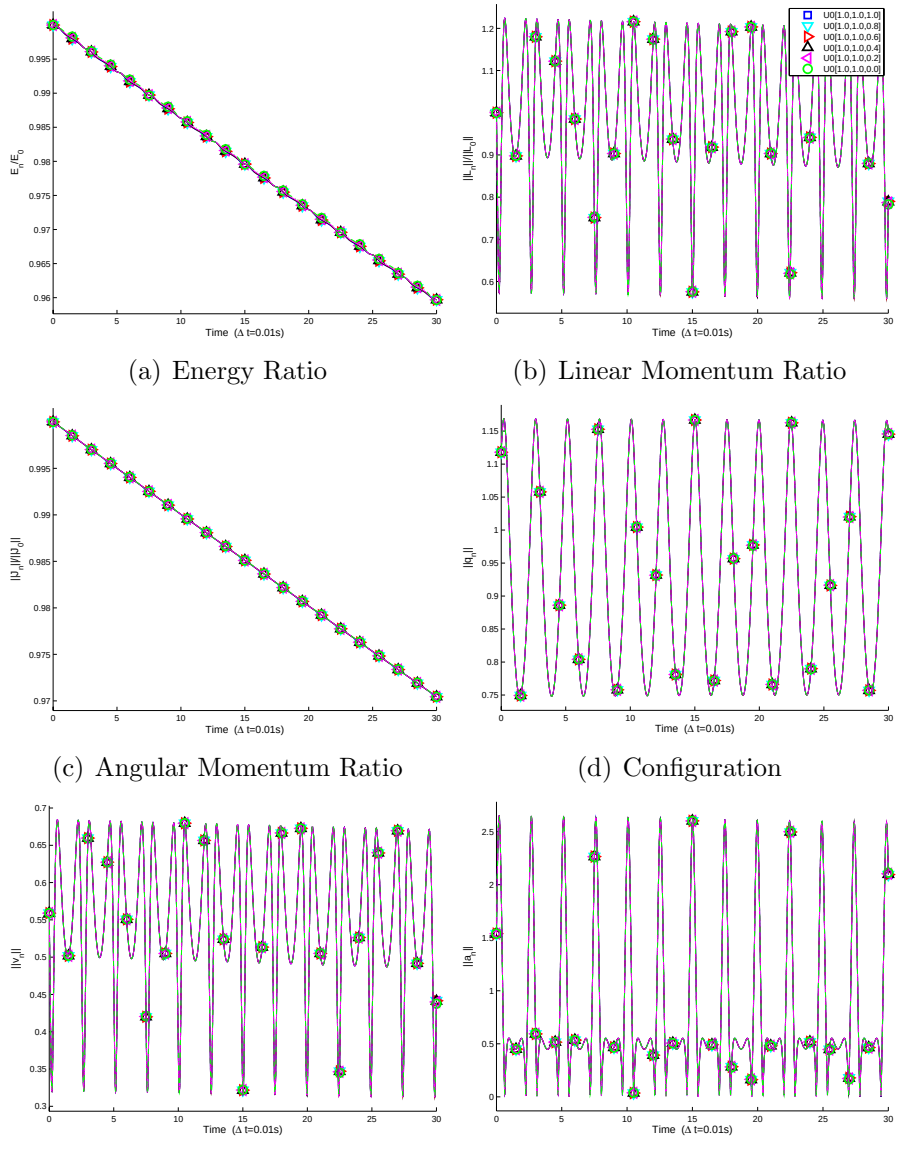


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 6.12: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Option II) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]

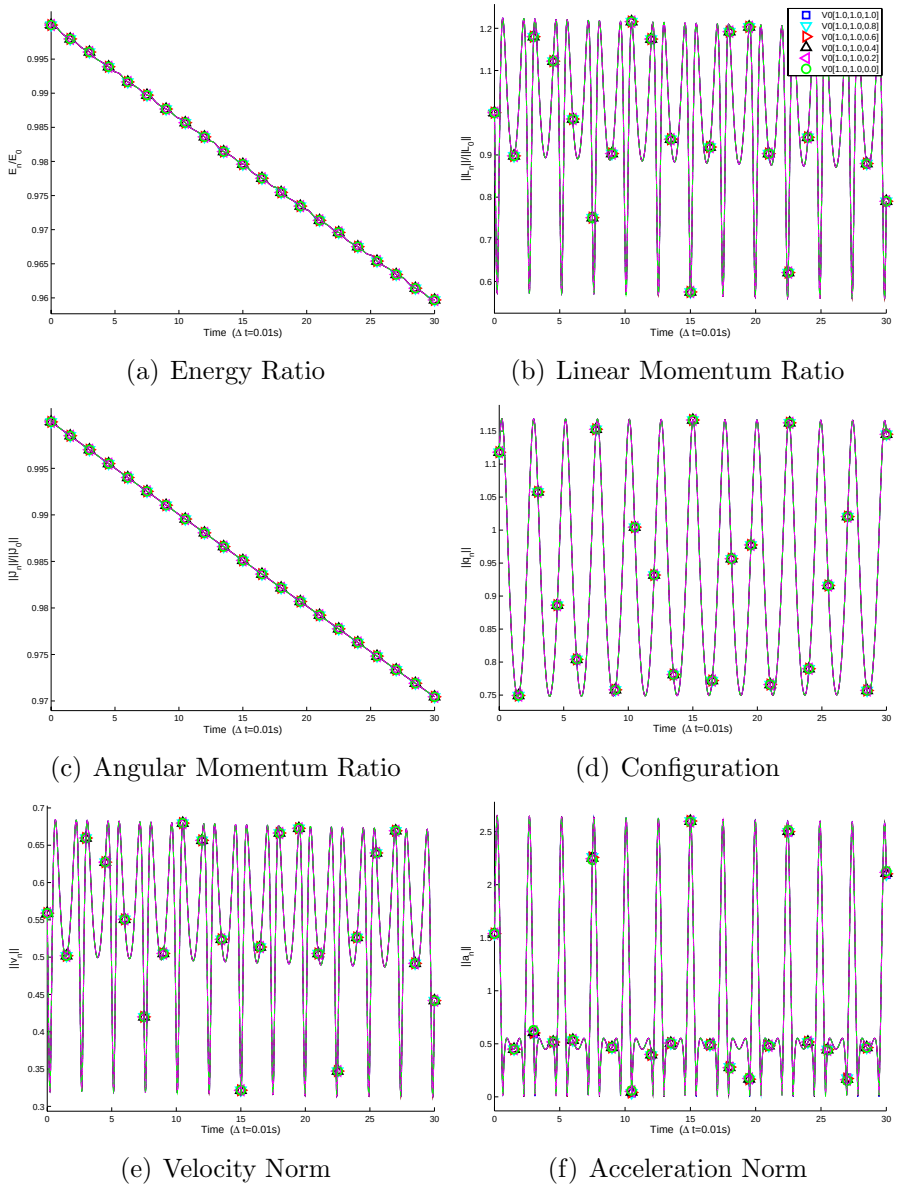


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 6.13: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Option II) - U0V0(1.0,1.0,ρ_∞)]



(a) Energy Ratio

(b) Linear Momentum Ratio

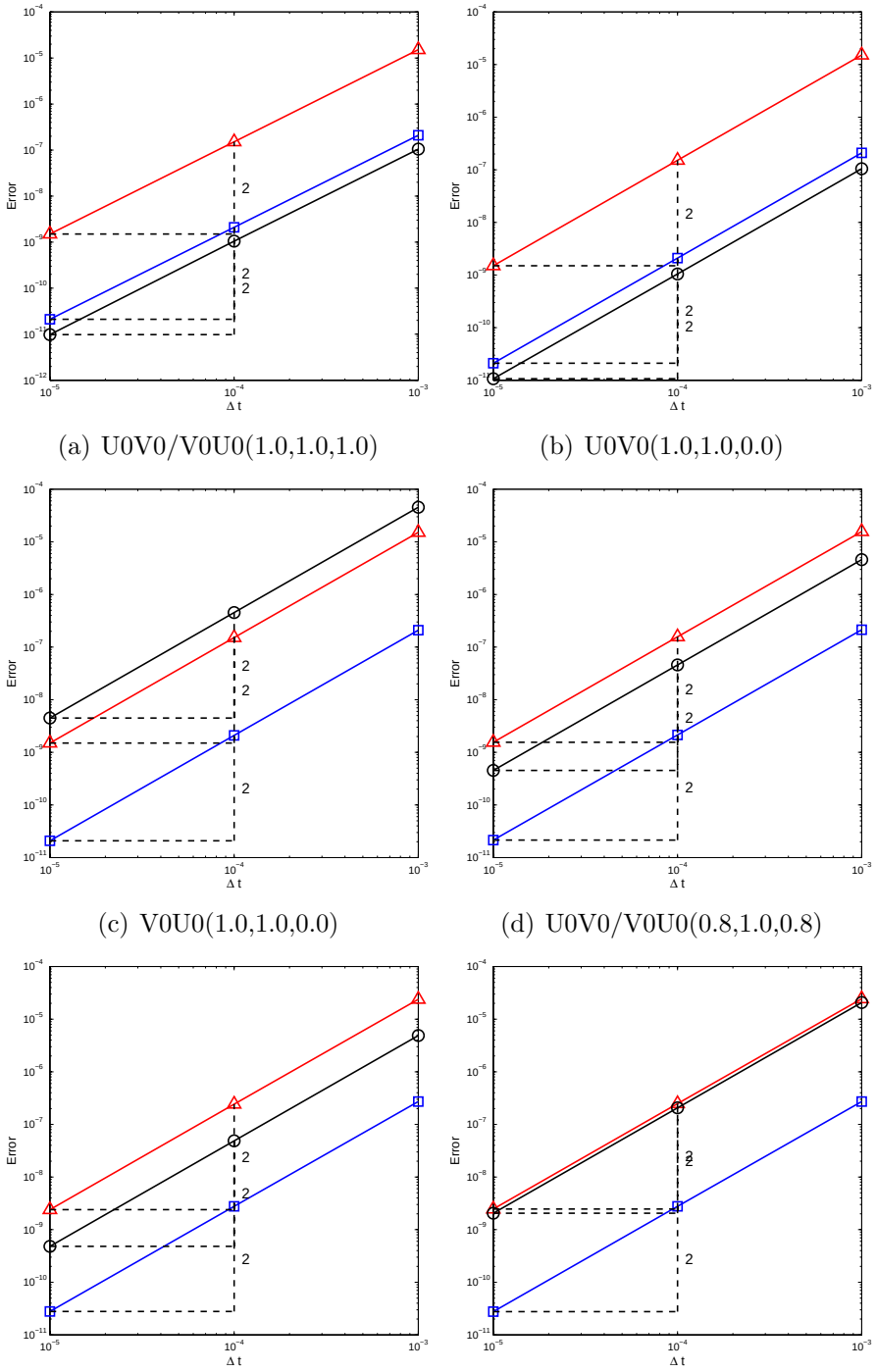
(c) Angular Momentum Ratio

(d) Configuration

(e) Velocity Norm

(f) Acceleration Norm

Figure 6.14: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Option II) - V0U0(1.0,1.0,ρ_∞)]



(a) U0V0/V0U0(1.0,1.0,1.0)

(b) U0V0(1.0,1.0,0.0)

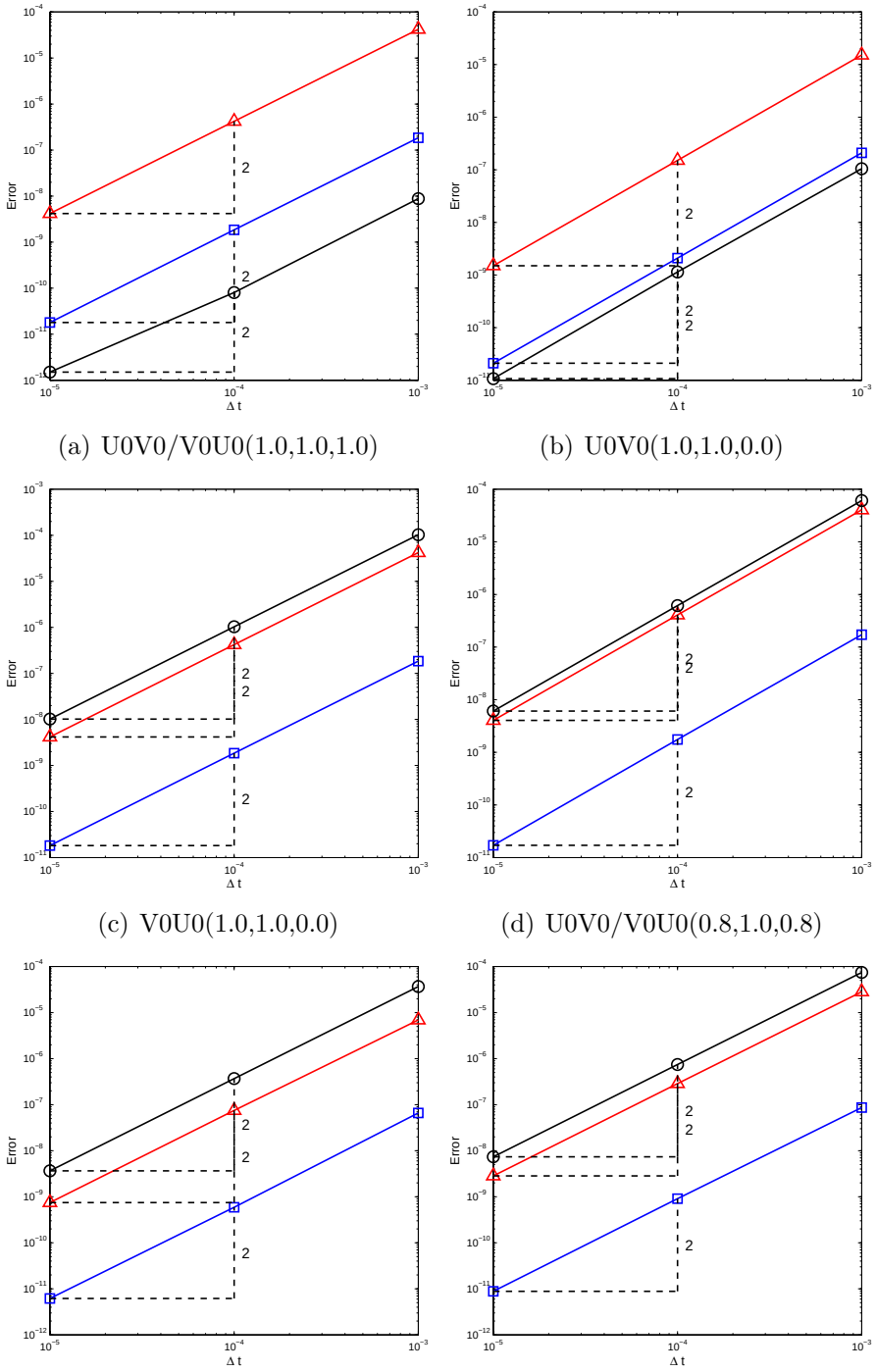
(c) V0U0(1.0,1.0,0.0)

(d) U0V0/V0U0(0.8,1.0,0.8)

(e) U0V1(0.6,0.9,0.2)

(f) V0U1(0.6,0.9,0.2)

Figure 6.15: Time accuracies in the configuration (□), velocity (△), and acceleration (○) in the forced mechanical system. [Problem: Kepler's problem] [Algorithm: Implicit GSSSS family of algorithms (Option I)]



(a) U0V0/V0U0(1.0,1.0,1.0)

(b) U0V0(1.0,1.0,0.0)

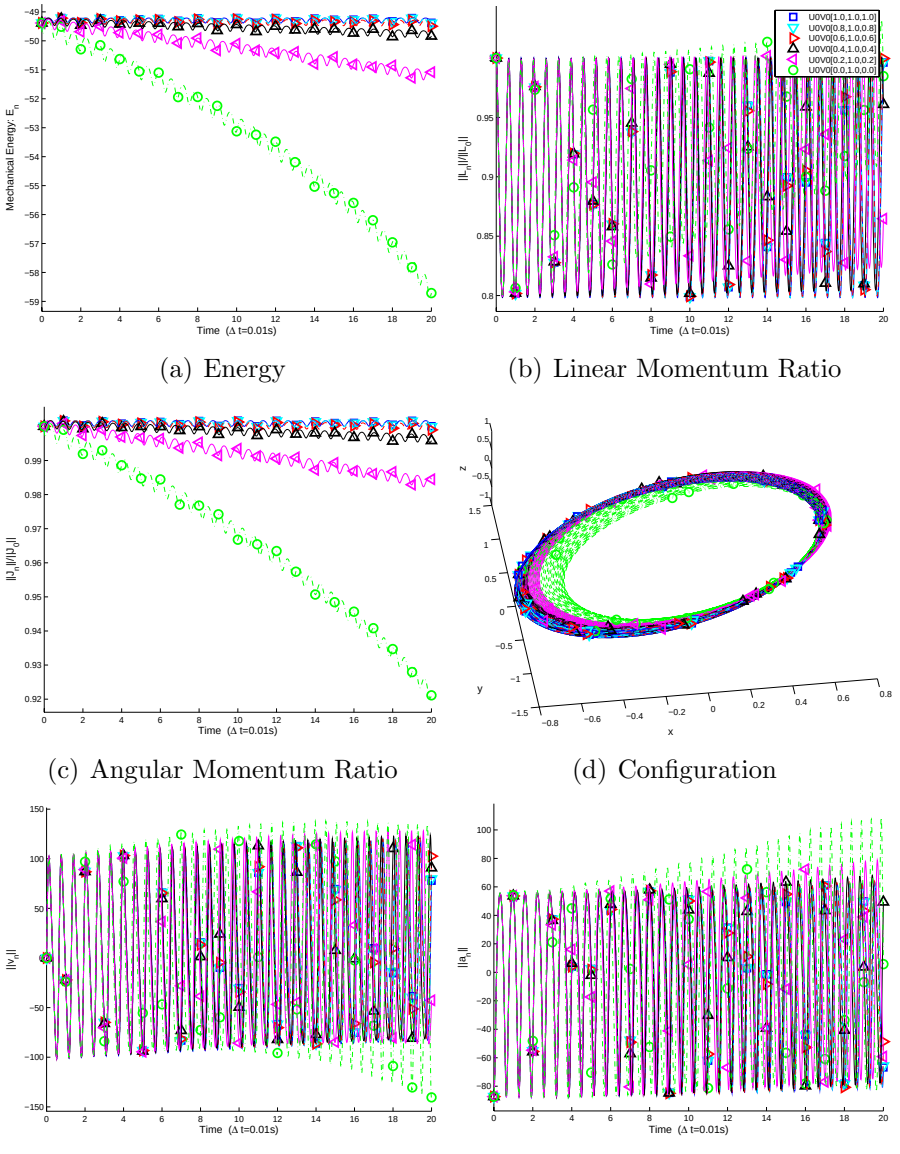
(c) V0U0(1.0,1.0,0.0)

(d) U0V0/V0U0(0.8,1.0,0.8)

(e) U0V1(0.6,0.9,0.2)

(f) V0U1(0.6,0.9,0.2)

Figure 6.16: Time accuracies in the configuration (□), velocity (△), and acceleration (○) in the forced mechani [Problem: Kepler's problem] [Algorithm: Implicit GSSSS family of algorithms (Option II)]

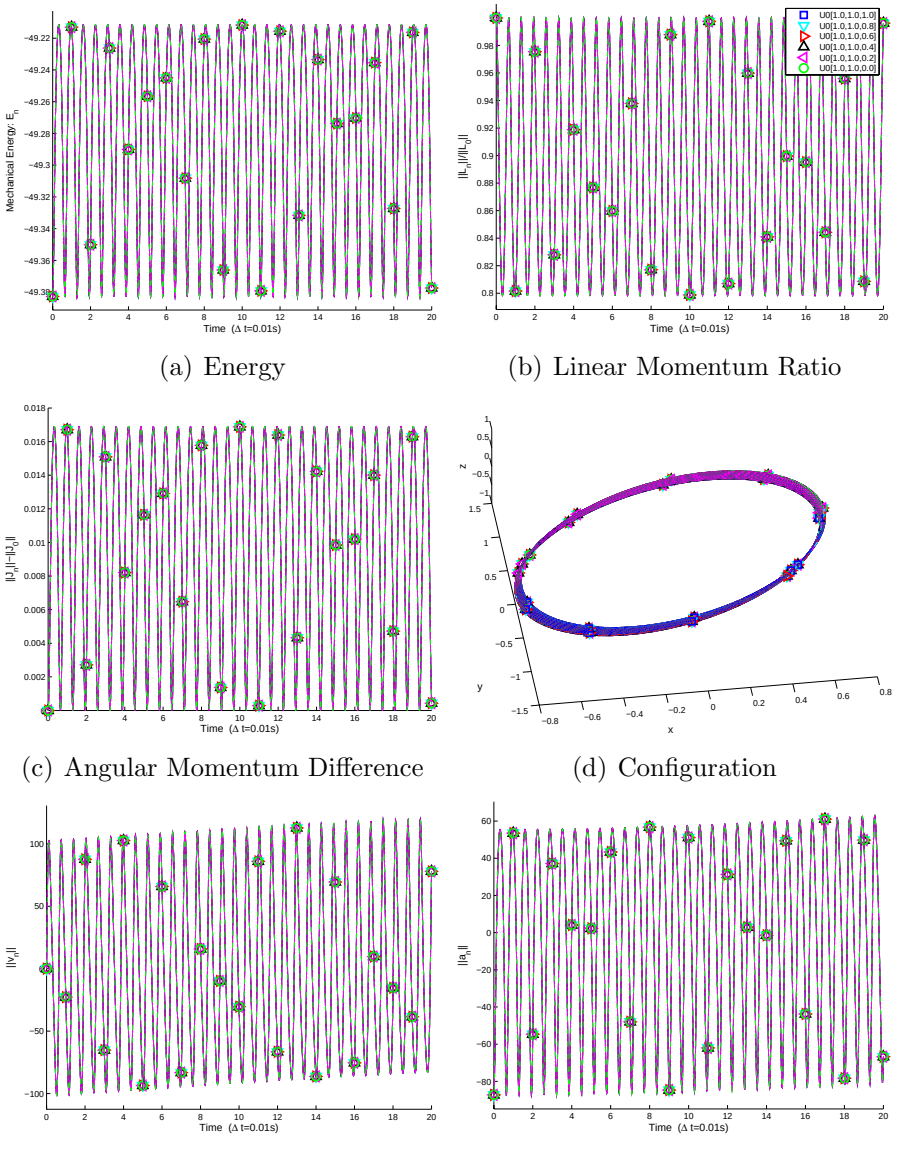


(a) Energy (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 6.17: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Implicit GSSSS family of algorithms (Option I) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]

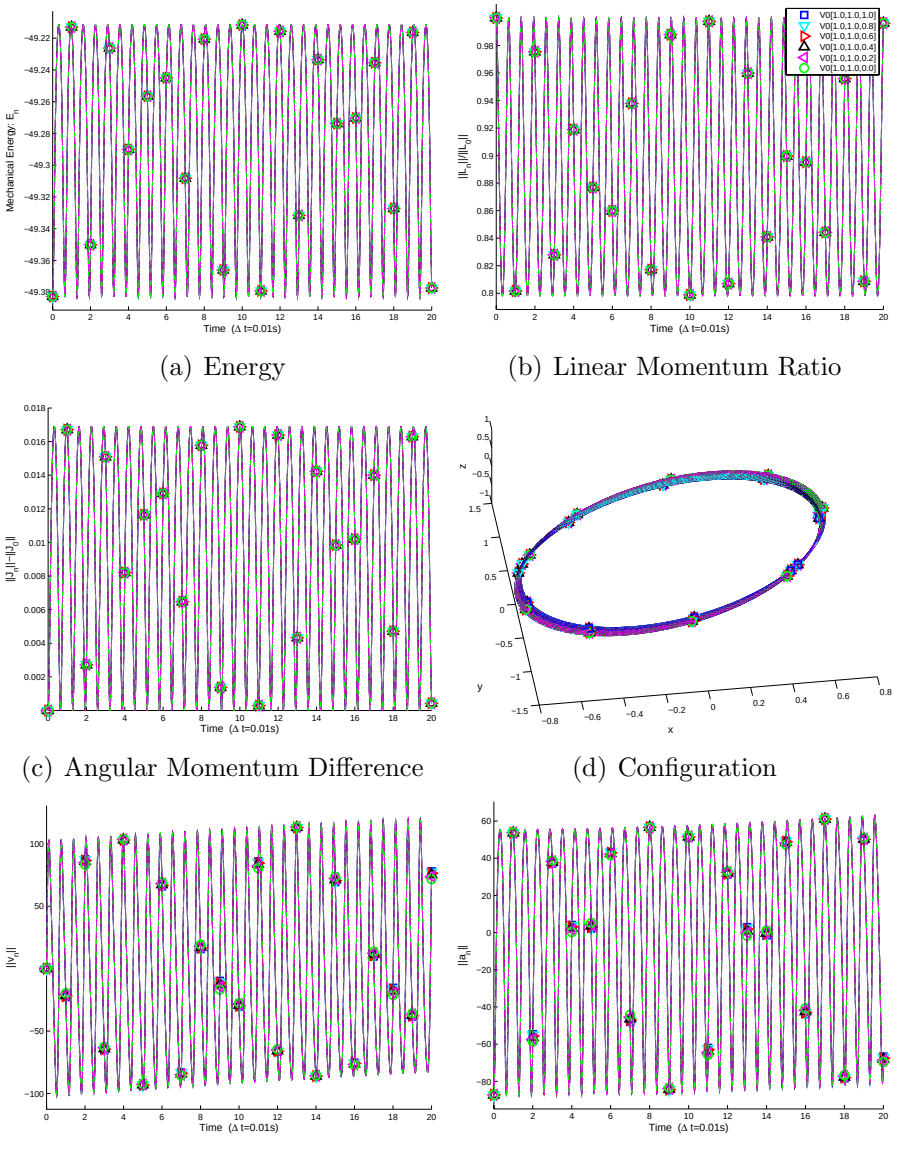


(a) Energy (b) Linear Momentum Ratio

(c) Angular Momentum Difference (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 6.18: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Implicit GSSSS family of algorithms (Option I) - U0V0(1.0,1.0,ρ_∞)]

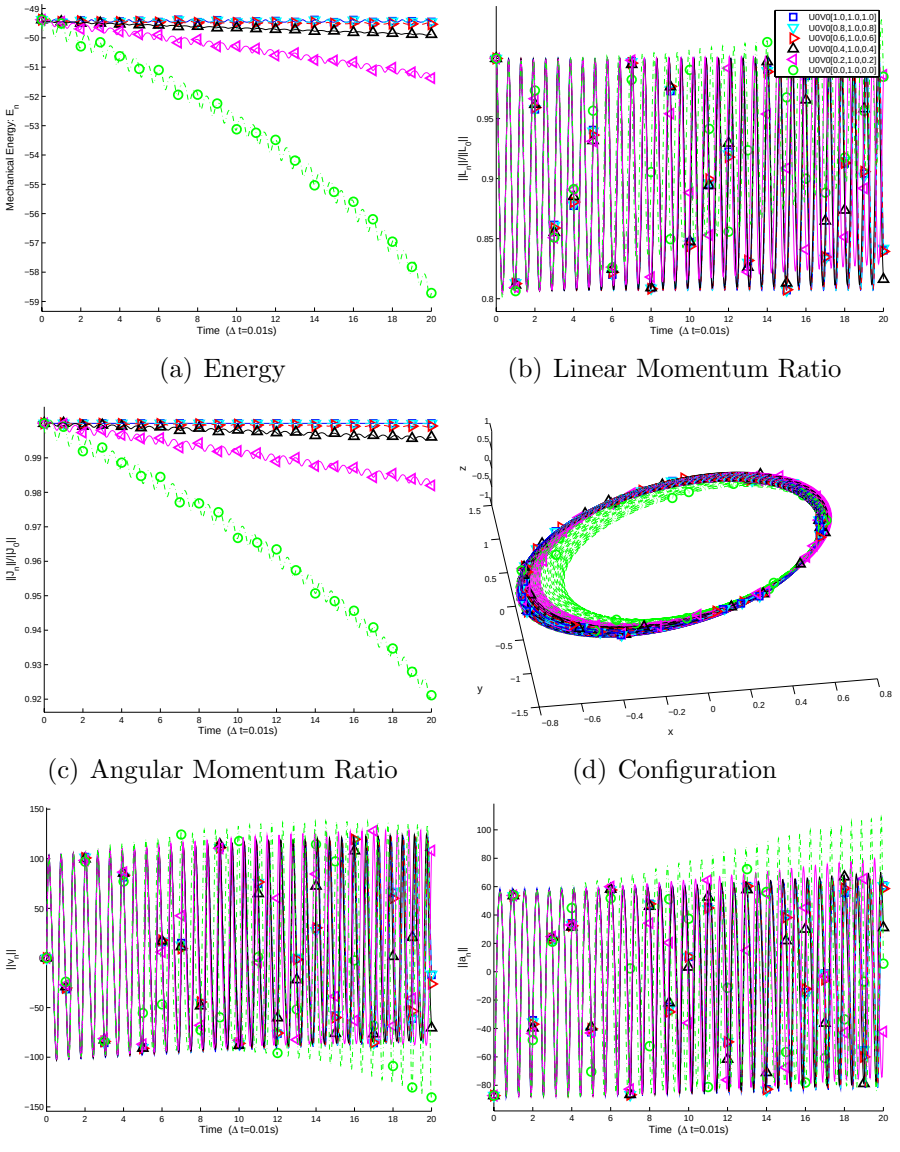


(a) Energy (b) Linear Momentum Ratio

(c) Angular Momentum Difference (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 6.19: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Implicit GSSSS family of algorithms (Option I) - V0U0(1.0,1.0,ρ_∞)]

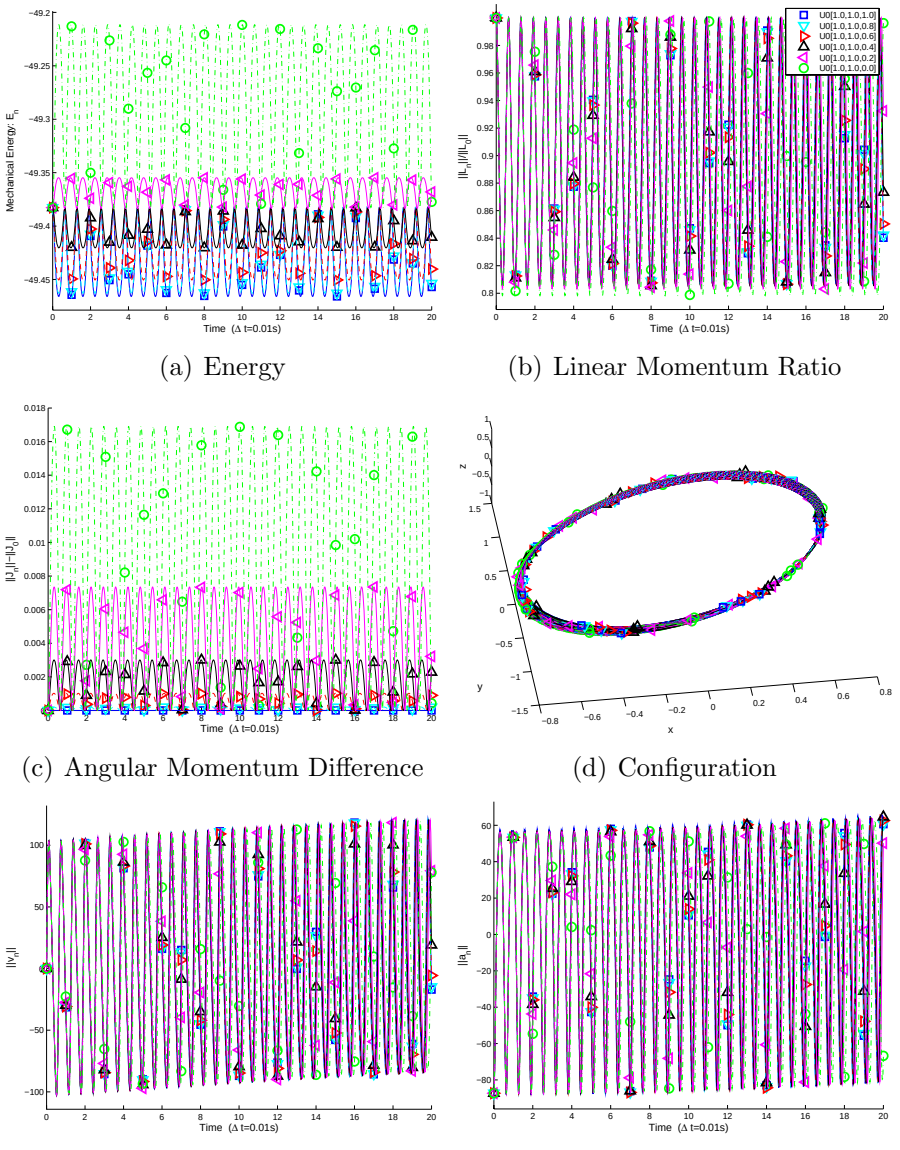


(a) Energy (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 6.20: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Implicit GSSSS family of algorithms (Option II) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]

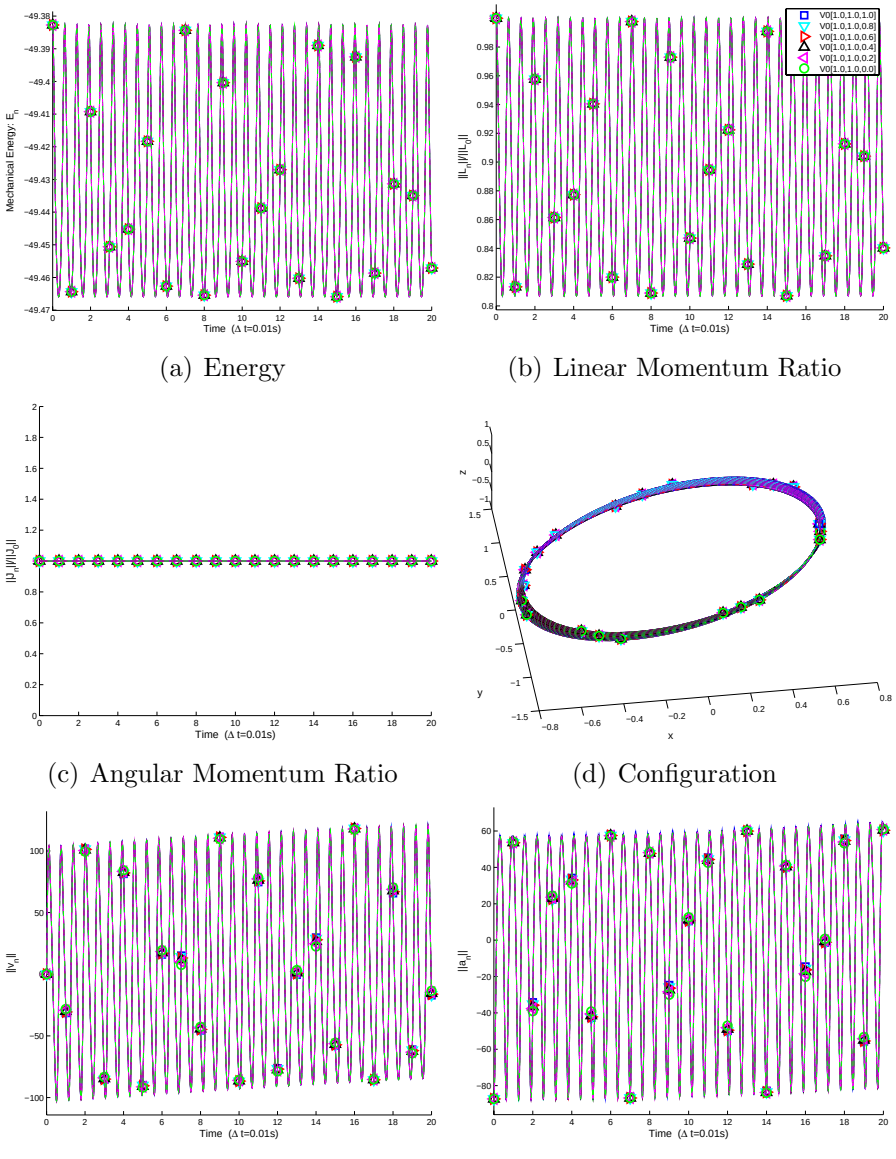


(a) Energy (b) Linear Momentum Ratio

(c) Angular Momentum Difference (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 6.21: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Implicit GSSSS family of algorithms (Option II) - U0V0(1.0,1.0,ρ_∞)]

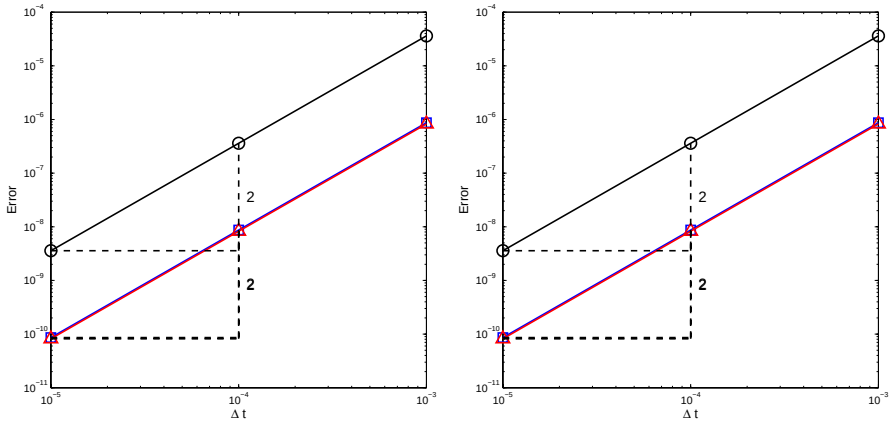


(a) Energy (b) Linear Momentum Ratio

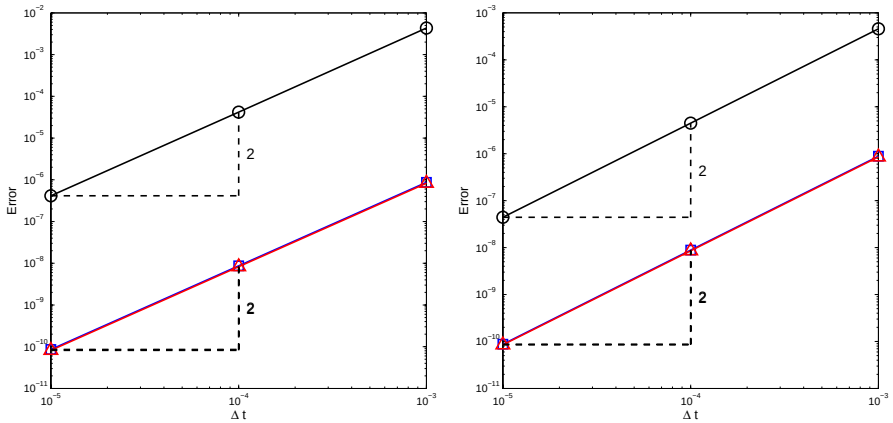
(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

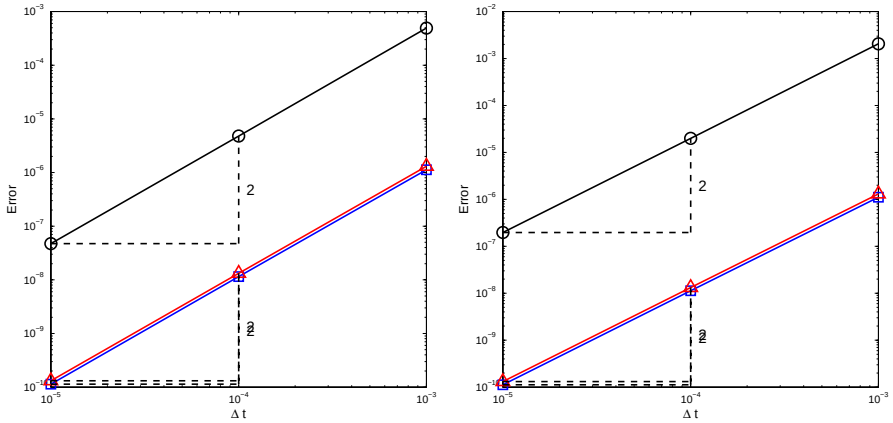
Figure 6.22: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Implicit GSSSS family of algorithms (Option II) - V0U0(1.0,1.0,ρ_∞)]



(a) U0V0/V0U0(1.0,1.0,1.0) (b) U0V0(1.0,1.0,0.0)

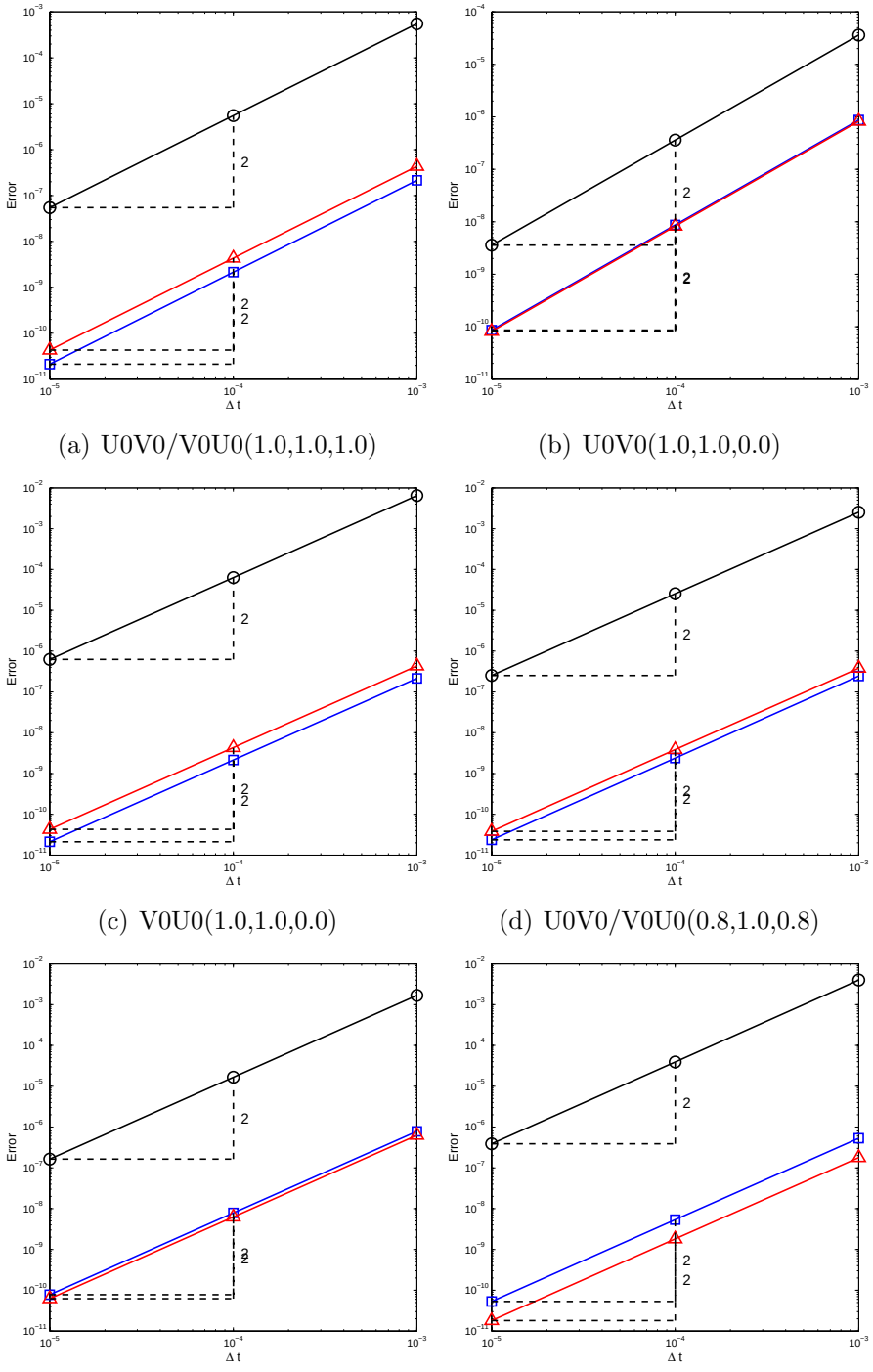


(c) V0U0(1.0,1.0,0.0) (d) U0V0/V0U0(0.8,1.0,0.8)



(e) U0V1(0.6,0.9,0.2) (f) V0U1(0.6,0.9,0.2)

Figure 6.23: Time accuracies in the configuration (□), velocity (△), and acceleration (○) in the forced mechanical system. [Problem: Lennard-Jones potential problem] [Algorithm: Implicit GSSSS family of algorithms (Option I)]



(a) U0V0/V0U0(1.0,1.0,1.0)

(b) U0V0(1.0,1.0,0.0)

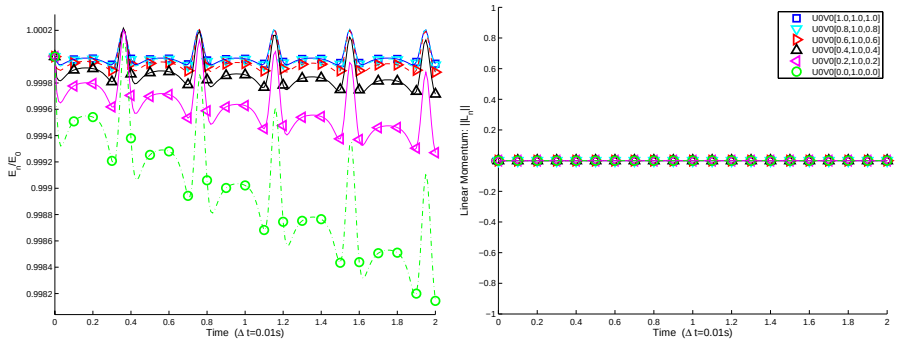
(c) V0U0(1.0,1.0,0.0)

(d) U0V0/V0U0(0.8,1.0,0.8)

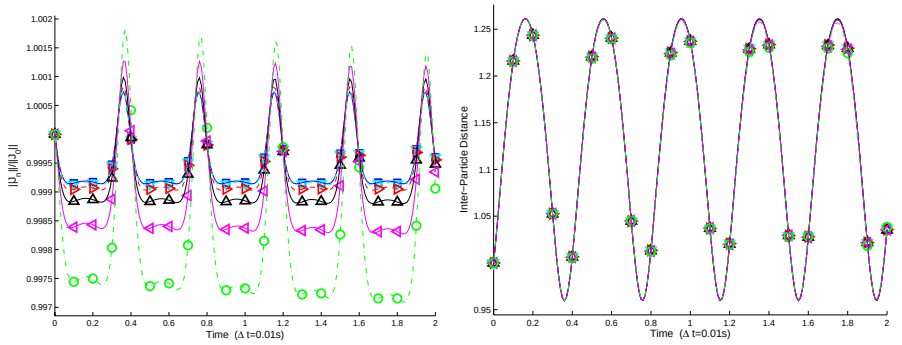
(e) U0V1(0.6,0.9,0.2)

(f) V0U1(0.6,0.9,0.2)

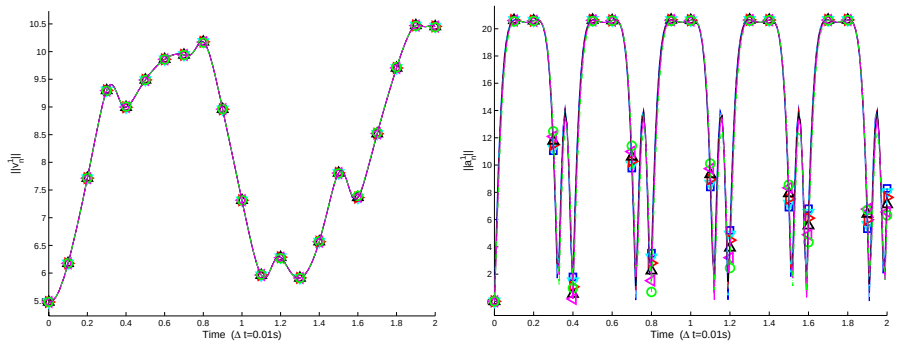
Figure 6.24: Time accuracies in the configuration (□), velocity (△), and acceleration (○) in the forced mechani [Problem: Lennard-Jones potential problem] [Algorithm: Implicit GSSSS family of algorithms (Option II)]



(a) Energy Ratio (b) Linear Momentum

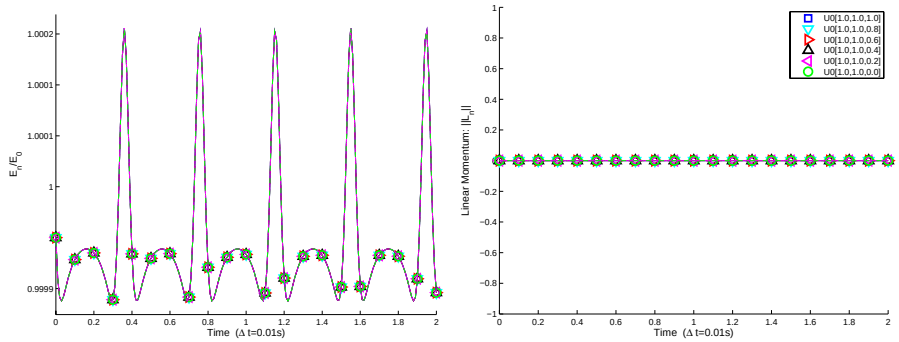


(c) Angular Momentum Ratio (d) Inter-particle Distance



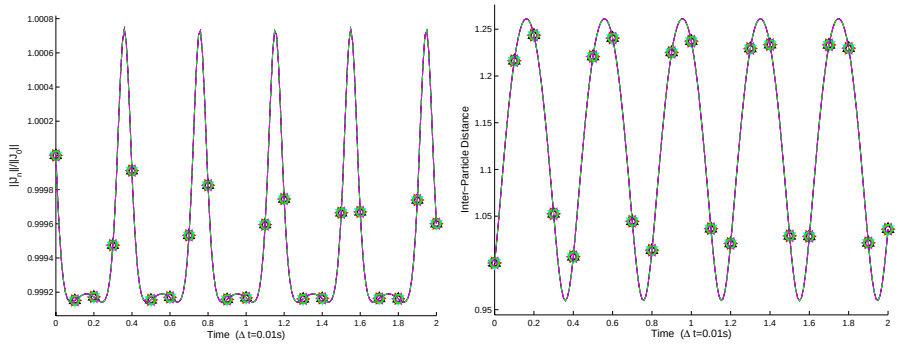
(e) Velocity Norm (f) Acceleration Norm

Figure 6.25: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Implicit GSSSS family of algorithms (Option I) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



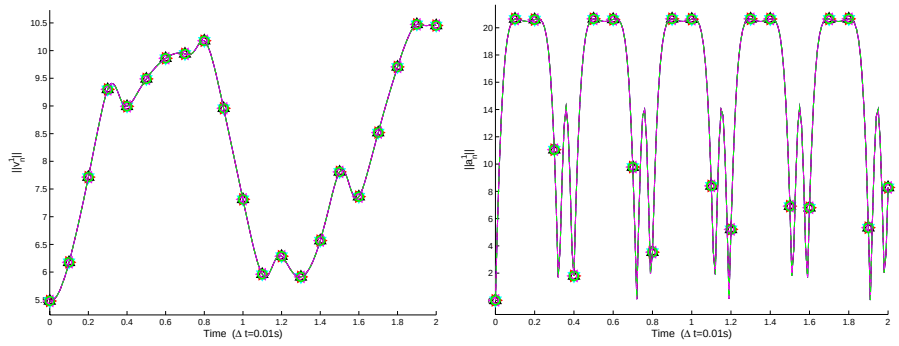
(a) Energy Ratio

(b) Linear Momentum



(c) Angular Momentum Ratio

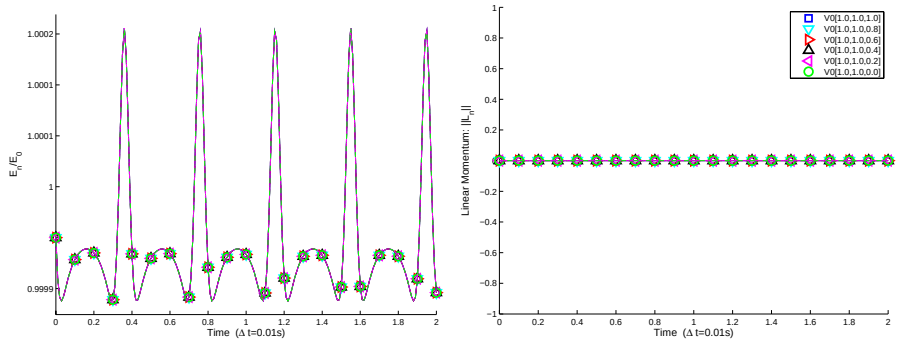
(d) Inter-particle Distance



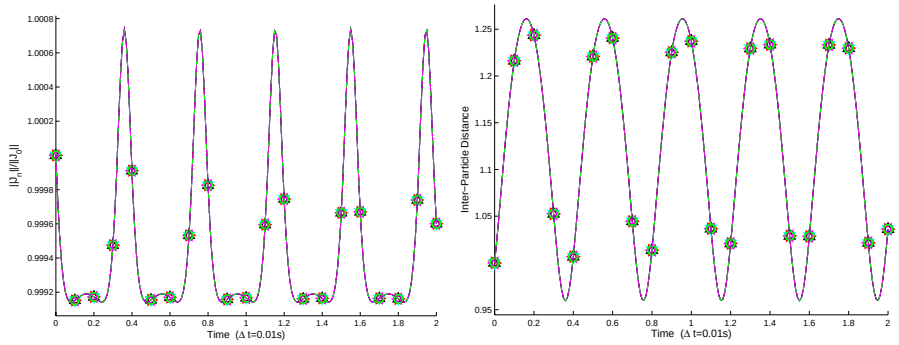
(e) Velocity Norm

(f) Acceleration Norm

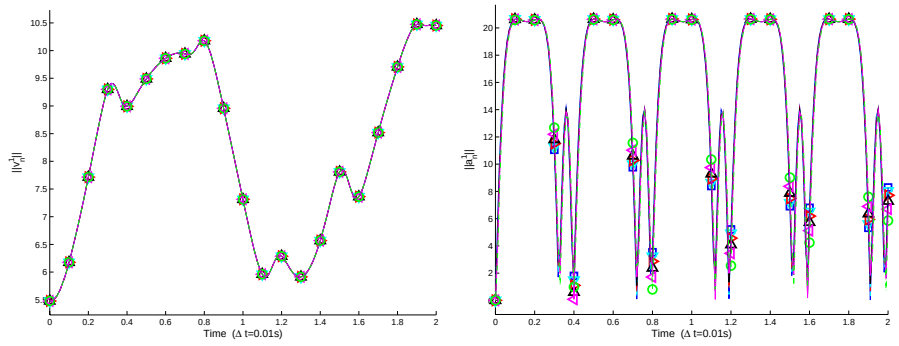
Figure 6.26: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Implicit GSSSS family of algorithms (Option I) - U0V0(1.0,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum

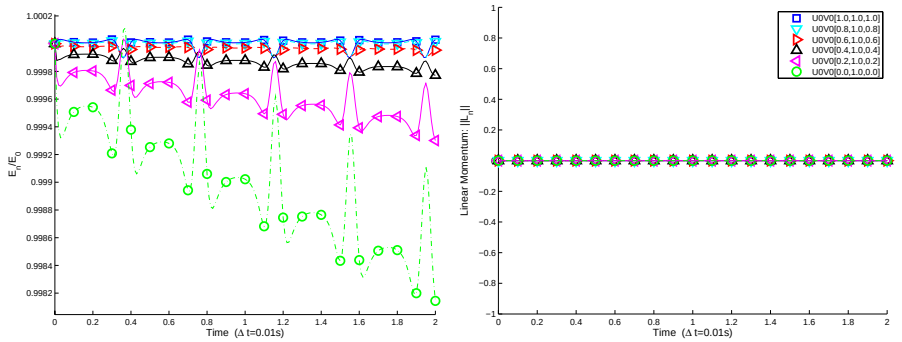


(c) Angular Momentum Ratio (d) Inter-particle Distance

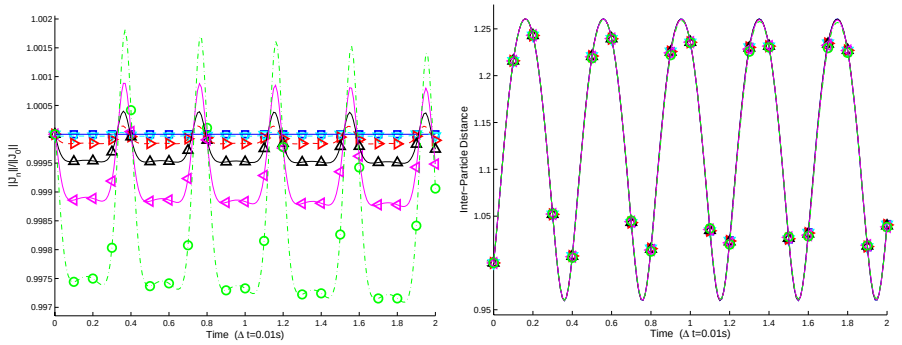


(e) Velocity Norm (f) Acceleration Norm

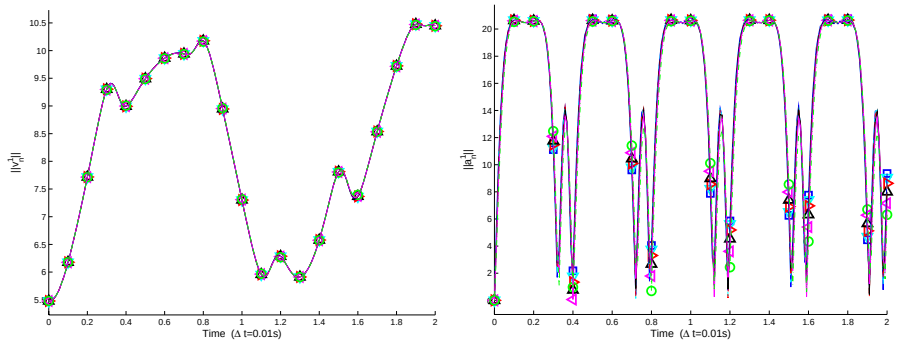
Figure 6.27: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Implicit GSSSS family of algorithms (Option I) - V0U0(1.0,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum

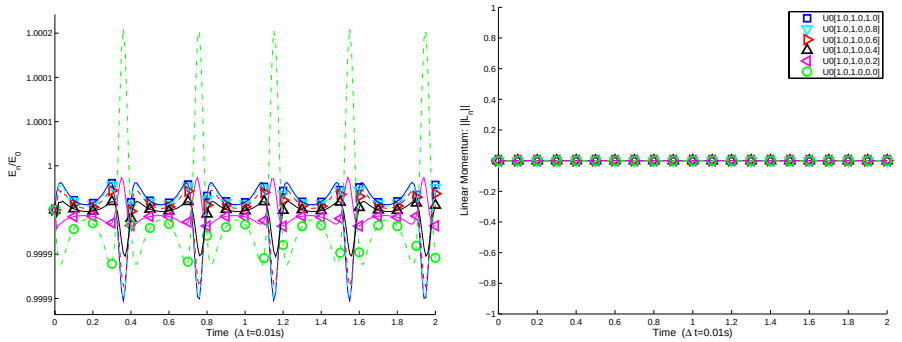


(c) Angular Momentum Ratio (d) Inter-particle Distance

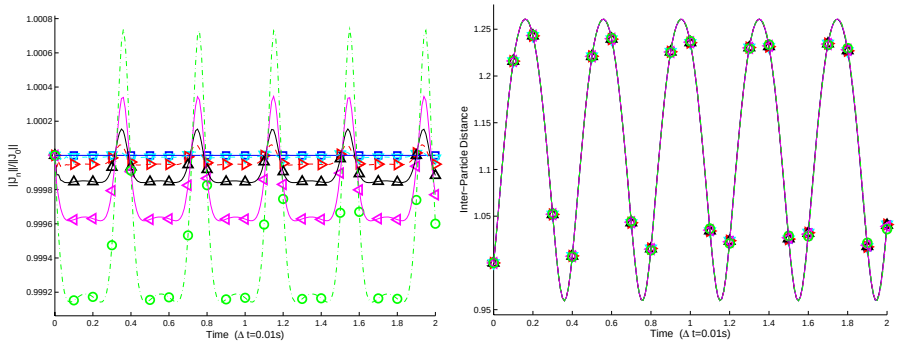


(e) Velocity Norm (f) Acceleration Norm

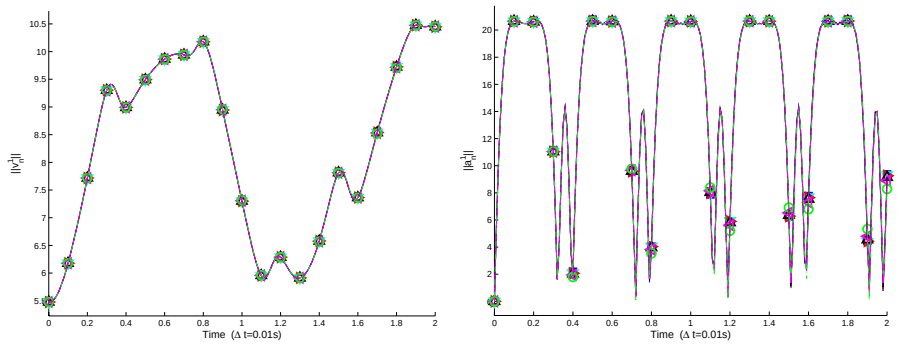
Figure 6.28: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Implicit GSSSS family of algorithms (Option II) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum

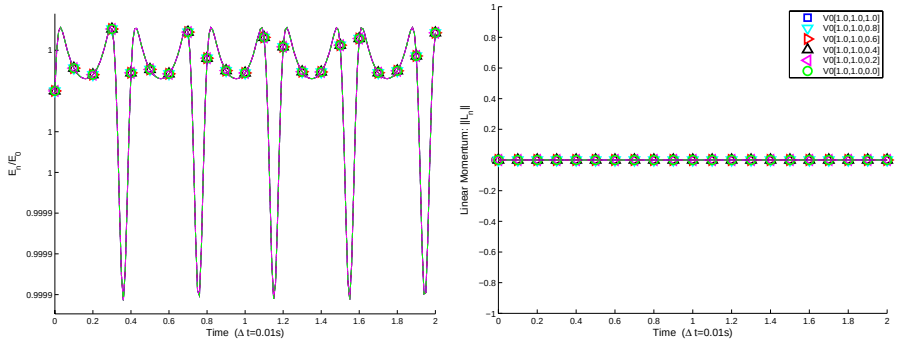


(c) Angular Momentum Ratio (d) Inter-particle Distance



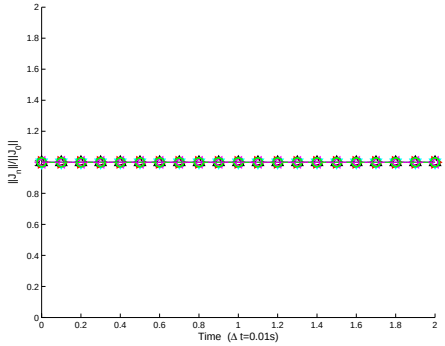
(e) Velocity Norm (f) Acceleration Norm

Figure 6.29: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Implicit GSSSS family of algorithms (Option II) - U0V0(1.0,1.0,ρ_∞)]

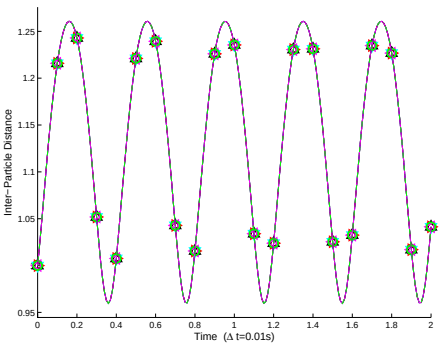


(a) Energy Ratio

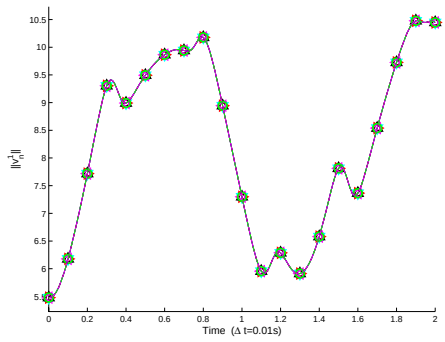
(b) Linear Momentum



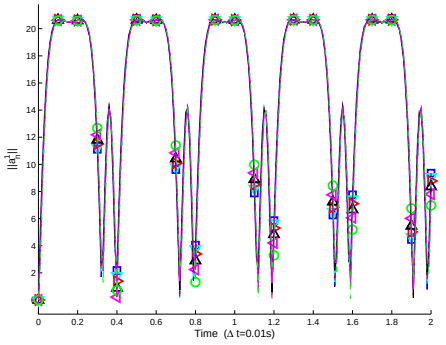
(c) Angular Momentum Ratio



(d) Inter-particle Distance



(e) Velocity Norm



(f) Acceleration Norm

Figure 6.30: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Implicit GSSSS family of algorithms (Option II) - V0U0(1.0,1.0,ρ_∞)]

Chapter 7

Relation between Exact Energy-Momentum Conserving Algorithms and Implicit Generalized Single Step Single Solve (I-GSSSS) Based Energy-Momentum Conserving Algorithms and Designs

7.1 N-body System

Exact energy and energy-momentum conserving and dissipative implicit algorithm designs in the two- and single-field forms for a system of N particles are shown in this chapter. We show various algorithm designs via two distinct approaches based

on the algorithmic time level consistency theorem, namely, the discrete total energy conservation method, and the classical and normalized time weighted residual methodologies; and therein draw comparisons among the various time-stepping algorithms and features. The relation between the energy and energy-momentum conserving time-stepping algorithms derived naturally from the principle of conservation of discrete mechanical energy by means of the mean value theorem and the implicit GSSSS family of algorithms that are alternately derived by means of the classical/normalized time weighted residual methodology is also emphasized. This is to enable conclusions of energy-momentum conserving features to be drawn.

7.1.1 Two-field Form: Discrete Total Energy Framework in a Conservative System: Implicit Energy and Energy-Momentum Conserving Algorithms

Exact Energy Conserving Implicit Algorithms

Consider the time discretization of the equation of motion in a conservative dynamical system of N particles for a time interval $\mathbb{I} = [t_0, t_f]$ split into subintervals, i.e., $\mathbb{I} = [t_0, t_f = t_0 + f\Delta t] = \bigcup_{n=0}^{f-1}[t_n, t_{n+1}]$, $n \in \{0, 1, 2, \cdots, f-1\}$. We assume that the system is holonomic-scleronomic. The principle of conservation of the total energy in a mechanical system free from constraints for the nondead load case ($\mathscr{P}^{\mathrm{D}}_{\mathrm{input}} = \mathscr{P}^{\mathrm{F}}_{\mathrm{input}} = 0$) together with the kinematic constraint and the initial

conditions are summarized as follows:

$$\boxed{\begin{aligned}&\textbf{\textit{Principle of balance of mechanical energy:}}\\&\frac{d\mathscr{E}(\mathbf{q}(t),\boldsymbol{\nu}(t))}{dt}=0\quad\forall t\in\mathbb{I}\\&\textbf{\textit{Kinematic constraint:}}\\&\dot{\mathbf{q}}(t)=\boldsymbol{\nu}(t)\quad\forall t\in\mathbb{I}\\&\textbf{\textit{Initial conditions:}}\\&\mathbf{q}(t_0)=\mathbf{q}_0\\&\boldsymbol{\nu}(t_0)=\boldsymbol{\nu}_0\end{aligned}}\tag{7.1}$$

where the autonomous total energy is defined as the summation of the kinetic energy $\mathscr{T}(\boldsymbol{\nu})$ and the total potential energy $\mathscr{U}(\mathbf{q})$, i.e., $\mathscr{E}(\mathbf{q},\boldsymbol{\nu}) := \mathscr{T}(\boldsymbol{\nu})+\mathscr{U}(\mathbf{q}) = \mathscr{T}(\boldsymbol{\nu}) + \mathscr{U}^{\text{int}}(\mathbf{q}) + \mathscr{U}^{\text{c}}(\mathbf{q})$ in which

$$\mathscr{T}(\boldsymbol{\nu}) = \frac{1}{2}\boldsymbol{\nu}\cdot\mathbf{M}\boldsymbol{\nu};\ \ \mathscr{U}^{\text{int}}(\mathbf{q}) = \sum_{i=1}^{N}\sum_{j>i}^{N}\mathscr{U}_{ij}^{\text{int}}(\mathbf{q}^i,\mathbf{q}^j)\ \text{ and }\ \mathscr{U}^{\text{c}}(\mathbf{q}) = \sum_{i=1}^{N}\mathscr{U}_{i}^{\text{c}}(\mathbf{q})\tag{7.2}$$

Recall that $\mathbf{q}$ denote the generalized coordinates on a $3N$-dimensional linear, differentiable, and smooth configuration manifold $Q \equiv \mathbb{R}^{3N}$; namely, $\mathbf{q} \in Q \equiv \mathbb{R}^{3N}$. Hence, its total time derivative (velocity) is defined on the tangent space at $\mathbf{q} \in Q$ as $\dot{\mathbf{q}} := d\mathbf{q}/dt \in T_{\mathbf{q}}Q$. Note that the total potential energy is defined as the summation of the internal potential energy and the external potential energy, i.e., $\mathscr{U}(\mathbf{q}) = \mathscr{U}^{\text{int}}(\mathbf{q}) + \mathscr{U}^{\text{c}}(\mathbf{q})$. We assume that the symmetric mass matrix $\mathbf{M}$ is

constant. As shown before, Eq. (7.1) yields

$$\boxed{\begin{array}{l} \textbf{\textit{Balance equation:}} \\ \dfrac{d}{dt}\left(\dfrac{\partial \mathscr{E}(\mathbf{q},\boldsymbol{\nu})}{\partial \boldsymbol{\nu}}\right) + \dfrac{\partial \mathscr{E}(\mathbf{q},\boldsymbol{\nu})}{\partial \mathbf{q}} = \mathbf{0} \\ \text{or} \;\; \mathbf{M}\dot{\boldsymbol{\nu}}(t) + \nabla \mathscr{U}(\mathbf{q}(t)) = \mathbf{0} \quad \forall t \in \mathbb{I} \\ \textbf{\textit{Kinematic constraint:}} \\ \dot{\mathbf{q}}(t) = \boldsymbol{\nu}(t) \quad \forall t \in \mathbb{I} \\ \textbf{\textit{Initial conditions:}} \\ \mathbf{q}(t_0) = \mathbf{q}_0 \\ \boldsymbol{\nu}(t_0) = \boldsymbol{\nu}_0 \end{array}} \tag{7.3}$$

Next, we start discretizing Eq. (7.1) via the ***mean value theorem*** to derive the exact energy conserving algorithm. The mean value theorem states: If the mechanical energy $\mathscr{E}(t) \in TQ$ is continuous on $[t_n, t_{n+1}]$ and differentiable on (t_n, t_{n+1}); then, there exists some $\gamma_0 \in (0,1)$ such that

$$\dot{\mathscr{E}}(t_{n+\gamma_0}) = \frac{\mathscr{E}(t_{n+1}) - \mathscr{E}(t_n)}{\Delta t} \tag{7.4}$$

In the conservative system, $\dot{\mathscr{E}}(t_{n+\gamma_0}) = 0$. The left-hand side of Eq. (7.4) can be written as

$$\begin{aligned} \dot{\mathscr{E}}(t_{n+\gamma_0}) &= \dot{\mathscr{T}}(\boldsymbol{\nu}(t_{n+\gamma_0})) + \dot{\mathscr{U}}(\mathbf{q}(t_{n+\gamma_0})) \\ &= \mathbf{M}\boldsymbol{\nu}(t_{n+\gamma_0}) \cdot \dot{\boldsymbol{\nu}}(t_{n+\gamma_0}) + \nabla \mathscr{U}(\mathbf{q}(t_{n+\gamma_0})) \cdot \dot{\mathbf{q}}(t_{n+\gamma_0}) \\ &= \mathbf{M}\dot{\boldsymbol{\nu}}(t_{n+\gamma_0}) \cdot \boldsymbol{\nu}(t_{n+\gamma_0}) + \nabla \mathscr{U}(\mathbf{q}(t_{n+\gamma_0})) \cdot \dot{\mathbf{q}}(t_{n+\gamma_0}) \end{aligned} \tag{7.5}$$

And the right-hand side of Eq. (7.4) can be written as

$$\frac{\mathscr{E}(t_{n+1}) - \mathscr{E}(t_n)}{\Delta t} = \mathbf{M}\frac{\boldsymbol{\nu}(t_{n+1}) - \boldsymbol{\nu}(t_n)}{\Delta t} \cdot \frac{\boldsymbol{\nu}(t_{n+1}) + \boldsymbol{\nu}(t_n)}{2} + \frac{\mathscr{U}(\mathbf{q}(t_{n+1})) - \mathscr{U}(\mathbf{q}(t_n))}{\Delta t} \tag{7.6}$$

If $\mathscr{U}(\mathbf{q})$ is continuous on $[\mathbf{q}(t_n), \mathbf{q}(t_{n+1})]$ and differentiable on $(\mathbf{q}(t_n), \mathbf{q}(t_{n+1}))$; then, we have from the mean value theorem that there exists some $\beta_0 \in (0,1)$

such that

$$\mathscr{U}(\mathbf{q}(t_{n+1})) - \mathscr{U}(\mathbf{q}(t_n)) = [\mathbf{q}(t_{n+1}) - \mathbf{q}(t_n)] \cdot \nabla \mathscr{U}(\mathbf{q}_{n+\beta_0}) \tag{7.7}$$

where

$$\mathbf{q}_{n+\beta_0} := [1-\beta_0]\,\mathbf{q}_n + \beta_0 \mathbf{q}_n \in Q \tag{7.8}$$

Note $\mathbf{q}_{n+\beta_0} \neq \mathbf{q}(t_{n+\beta_0})$ for $\beta_0 \in (0,1)$ in general, and $\mathbf{q}(t_n) \approx \mathbf{q}_n \in Q$ and $\mathbf{q}(t_{n+1}) \approx \mathbf{q}_{n+1} \in Q$. Thus, Eq. (7.6) can be written as

$$\begin{aligned} \frac{\mathscr{E}(t_{n+1}) - \mathscr{E}(t_n)}{\Delta t} &= \mathbf{M}\frac{\boldsymbol{\nu}(t_{n+1}) - \boldsymbol{\nu}(t_n)}{\Delta t} \cdot \frac{\boldsymbol{\nu}(t_{n+1}) + \boldsymbol{\nu}(t_n)}{2} \\ &\quad + \nabla \mathscr{U}(\mathbf{q}_{n+\beta_0}) \cdot \frac{\mathbf{q}(t_{n+1}) - \mathbf{q}(t_n)}{\Delta t} \end{aligned} \tag{7.9}$$

Hence, the conservation of the total energy at time $t = t_{n+\gamma_0} \in [t_n, t_{n+1}]$ yields

$$\begin{aligned} 0 &= \mathbf{M}\dot{\boldsymbol{\nu}}(t_{n+\gamma_0}) \cdot \boldsymbol{\nu}(t_{n+\gamma_0}) + \nabla \mathscr{U}(\mathbf{q}(t_{n+\gamma_0})) \cdot \dot{\mathbf{q}}(t_{n+\gamma_0}) \\ &= \mathbf{M}\frac{\boldsymbol{\nu}(t_{n+1}) - \boldsymbol{\nu}(t_n)}{\Delta t} \cdot \frac{\boldsymbol{\nu}(t_{n+1}) + \boldsymbol{\nu}(t_n)}{2} + \nabla \mathscr{U}(\mathbf{q}_{n+\beta_0}) \cdot \frac{\mathbf{q}(t_{n+1}) - \mathbf{q}(t_n)}{\Delta t} \end{aligned} \tag{7.10}$$

In Eq. (7.10), if we enforce

$$\boldsymbol{\nu}(t_{n+\gamma_0}) \cong \frac{\boldsymbol{\nu}(t_{n+1}) + \boldsymbol{\nu}(t_n)}{2} \quad \text{and} \quad \dot{\mathbf{q}}(t_{n+\gamma_0}) \cong \frac{\mathbf{q}(t_{n+1}) - \mathbf{q}(t_n)}{\Delta t} \tag{7.11}$$

we get

$$\frac{\mathbf{q}(t_{n+1}) - \mathbf{q}(t_n)}{\Delta t} \cong \frac{\boldsymbol{\nu}(t_{n+1}) + \boldsymbol{\nu}(t_n)}{2} \tag{7.12}$$

due to the kinematic constraint at time $t_{n+\gamma_0}$, $\dot{\mathbf{q}}(t_{n+\gamma_0}) = \boldsymbol{\nu}(t_{n+\gamma_0}) \in T_{\mathbf{q}}Q$. Notice that Eq. (7.12) implies the second-order time approximation between $\mathbf{q}$ and $\boldsymbol{\nu}$ in the fully-discrete system. From Eq. (7.11) and Eq. (7.12), the algorithmic time level of the kinematic constraint is at time $t_{n+1/2}$. The corresponding fully-discrete equation of motion is given by

$$\mathbf{M}\frac{\boldsymbol{\nu}(t_{n+1}) - \boldsymbol{\nu}(t_n)}{\Delta t} + \nabla \mathscr{U}(\mathbf{q}_{n+\beta_0}) \cong \mathbf{0} \tag{7.13}$$

from Eq. (7.10) due to the enforcement, Eq. (7.11). Hence, the following algorithm is obtained with the given initial conditions ($\mathbf{q}_0 = \mathbf{q}(t_0)$ and $\boldsymbol{\nu}_0 = \boldsymbol{\nu}(t_0)$):

$$\begin{aligned} &\mathbf{M}\frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t} + \nabla \mathscr{U}(\mathbf{q}_{n+\beta_0}) \cong \mathbf{0} \\ &\frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t} \cong \frac{\boldsymbol{\nu}_{n+1} + \boldsymbol{\nu}_n}{2} \end{aligned} \tag{7.14}$$

with

$$\mathscr{U}(\mathbf{q}_{n+1}) - \mathscr{U}(\mathbf{q}_n) = [\mathbf{q}_{n+1} - \mathbf{q}_n] \cdot \nabla \mathscr{U}(\mathbf{q}_{n+\beta_0}) \tag{7.15}$$

where $\square_{n+1} \approx \square(t_{n+1})$ and $\square_n \approx \square(t_n)$ for $n \in \{0, 1, 2, \cdots, f\}$. The exact energy conservation within a time step $[t_n, t_{n+1}]$ in the sense of $\mathscr{E}_{n+1} = \mathscr{E}_n$ in which $\mathscr{E}_{n+1} := \mathscr{E}(\mathbf{q}_{n+1}, \boldsymbol{\nu}_{n+1}) \approx \mathscr{E}(t_{n+1})$ and $\mathscr{E}_n := \mathscr{E}(\mathbf{q}_n, \boldsymbol{\nu}_n) \approx \mathscr{E}(t_n)$ is naturally guaranteed. Next, we study the time accuracy of the exact energy conserving algorithm given by Eq. (7.14).

Accuracy

The order of accuracy of the algorithm above is two for any $\beta_0 \in (0, 1)$. Let the local truncation error of the algorithm be defined as $\boldsymbol{\tau}(\Delta t) = (\boldsymbol{\tau}_1(\Delta t), \boldsymbol{\tau}_2(\Delta t))^T$ where

$$\boldsymbol{\tau}_1(\Delta t) := \mathbf{M}[\boldsymbol{\nu}(t_{n+1}) - \boldsymbol{\nu}(t_n)] + \Delta t \bar{\mathbf{f}}^{\text{con}} \equiv \mathbf{0} \tag{7.16}$$

$$\boldsymbol{\tau_2}(\Delta t) := \mathbf{q}(t_{n+1}) - \mathbf{q}(t_n) - \frac{\Delta t}{2}[\boldsymbol{\nu}(t_{n+1}) + \boldsymbol{\nu}(t_n)] \equiv \mathbf{0} \tag{7.17}$$

in which $\bar{\mathbf{f}}^{\text{con}}$ is the exact value of $\nabla \mathscr{U}(\mathbf{q}_{n+\beta_0})$. Since the Taylor series expansions of $\mathscr{U}(\mathbf{q}_{n+1})$ and $\mathscr{U}(\mathbf{q}_n)$ about $\mathbf{q}_{n+\beta_0}$ yields

$$\begin{aligned} \mathscr{U}(\mathbf{q}_{n+1}) &= \mathscr{U}(\mathbf{q}_{n+\beta_0}) + [1 - \beta_0]\Delta \mathbf{q} \cdot \nabla \mathscr{U}(\mathbf{q}_{n+\beta_0}) \\ &\quad + \frac{[1 - \beta_0]^2}{2}\Delta \mathbf{q} \cdot \nabla^2 \mathscr{U}(\mathbf{q}_{n+\beta_0})\Delta \mathbf{q} + \mathcal{O}(\Delta t^3) \end{aligned} \tag{7.18}$$

$$\begin{aligned} \mathscr{U}(\mathbf{q}_n) &= \mathscr{U}(\mathbf{q}_{n+\beta_0}) - \beta_0 \Delta \mathbf{q} \cdot \nabla \mathscr{U}(\mathbf{q}_{n+\beta_0}) \\ &\quad + \frac{\beta_0^2}{2}\Delta \mathbf{q} \cdot \nabla^2 \mathscr{U}(\mathbf{q}_{n+\beta_0})\Delta \mathbf{q} + \mathcal{O}(\Delta t^3) \end{aligned} \tag{7.19}$$

respectively, we have

$$\mathscr{U}(\mathbf{q}_{n+1}) - \mathscr{U}(\mathbf{q}_n) = \Delta\mathbf{q} \cdot \left[\nabla\mathscr{U}(\mathbf{q}_{n+\beta_0}) + \frac{1-2\beta_0}{2}\nabla^2\mathscr{U}(\mathbf{q}_{n+\beta_0})\Delta\mathbf{q}\right] + \mathcal{O}(\Delta t^3) \tag{7.20}$$

Notice that $\Delta\mathbf{q}$ can be approximated by the Taylor series expansion at time t_n as

$$\Delta\mathbf{q} \approx \mathbf{q}(t_{n+1}) - \mathbf{q}(t_n) = \mathbf{q}(t_n) + \Delta t\dot{\mathbf{q}}(t_n) + \frac{\Delta t^2}{2}\ddot{\mathbf{q}}(t_n) + \mathcal{O}(\Delta t^3) \tag{7.21}$$

That is,

$$\begin{aligned}
\nabla\mathscr{U}(\mathbf{q}_{n+\beta_0}) &= \nabla\mathscr{U}(\mathbf{q}_{n+\beta_0}) + \frac{1-2\beta_0}{2}\nabla^2\mathscr{U}(\mathbf{q}_{n+\beta_0})\Delta\mathbf{q} + \mathcal{O}(\Delta t^2) \\
&= \nabla\mathscr{U}(\mathbf{q}(t_n)) + \beta_0\Delta t\nabla^2\mathscr{U}(\mathbf{q}_n)\dot{\mathbf{q}}(t_n) \\
&\quad + \frac{1-2\beta_0}{2}\Delta t\nabla^2\mathscr{U}(\mathbf{q}(t_n))\dot{\mathbf{q}}(t_n) + \mathcal{O}(\Delta t^2) \\
&= \nabla\mathscr{U}(\mathbf{q}(t_n)) + \frac{\Delta t}{2}\nabla^2\mathscr{U}(\mathbf{q}(t_n))\dot{\mathbf{q}}(t_n) + \mathcal{O}(\Delta t^2)
\end{aligned} \tag{7.22}$$

Substituting

$$\boldsymbol{\nu}(t_{n+1}) = \boldsymbol{\nu}(t_n) + \Delta t\dot{\boldsymbol{\nu}}(t_n) + \frac{\Delta t^2}{2}\ddot{\boldsymbol{\nu}}(t_n) + \mathcal{O}(\Delta t^3) \tag{7.23}$$

$$\begin{aligned}
\bar{\mathbf{f}}^{\text{con}} &\approx \nabla\mathscr{U}(\mathbf{q}_{n+\beta_0}) \\
&= \nabla\mathscr{U}(\mathbf{q}(t_n)) + \frac{\Delta t}{2}\nabla^2\mathscr{U}(\mathbf{q}(t_n))\dot{\mathbf{q}}(t_n) + \mathcal{O}(\Delta t^2)
\end{aligned} \tag{7.24}$$

into Eq. (7.65) yields

$$\begin{aligned}
\boldsymbol{\tau}_1(\Delta t) &= \Delta t\underbrace{\left[\mathbf{M}\dot{\boldsymbol{\nu}}(t_n) + \nabla\mathscr{U}(\mathbf{q}(t_n))\right]}_{=\mathbf{0}} \\
&\quad + \frac{\Delta t^2}{2}\underbrace{\left[\mathbf{M}\ddot{\boldsymbol{\nu}}(t_n) + \nabla^2\mathscr{U}(\mathbf{q}(t_n))\dot{\mathbf{q}}(t_n)\right]}_{=\mathbf{0}} + \mathcal{O}(\Delta t^3) \\
&= \mathcal{O}(\Delta t^3)
\end{aligned} \tag{7.25}$$

Similarly, substituting the Taylor series expansions of $\boldsymbol{\nu}(t_{n+1})$ and $\mathbf{q}(t_{n+1})$ at time t_n into Eq. (7.66) yields

$$\boldsymbol{\tau}_2(\Delta t) = \Delta t\underbrace{\left[\dot{\mathbf{q}}(t_n) - \boldsymbol{\nu}(t_n)\right]}_{=\mathbf{0}} + \frac{\Delta t^2}{2}\underbrace{\left[\ddot{\mathbf{q}}(t_n) - \dot{\boldsymbol{\nu}}(t_n)\right]}_{=\mathbf{0}} + \mathcal{O}(\Delta t^3) = \mathcal{O}(\Delta t^3) \tag{7.26}$$

due to the kinematic constraints. Hence,

$$\boldsymbol{\tau}(\Delta t) = \mathcal{O}(\Delta t^3) \;\text{ and }\; \| \boldsymbol{\tau}(\Delta t) \| = \mathcal{O}(\Delta t^3) \tag{7.27}$$

which shows the order of time accuracy of the algorithm is two for any $\beta_0 \in (0, 1)$.

Time Level Consistency

$$\dot{\boldsymbol{\nu}}(t^*_{\dot{\boldsymbol{\nu}}}) = \frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t} + \mathcal{O}(\Delta t^2) \tag{7.28}$$

with the algorithmic time level of $\dot{\boldsymbol{\nu}}$, i.e., $t^*_{\dot{\boldsymbol{\nu}}} := t_n + \alpha_{\dot{\boldsymbol{\nu}}}\Delta t$ for $\alpha_{\dot{\boldsymbol{\nu}}} \in [0, 1] \subset \mathbb{R}$; and the assumptions $\boldsymbol{\nu}_{n+1} \approx \boldsymbol{\nu}(t_{n+1})$ and $\boldsymbol{\nu}_n \approx \boldsymbol{\nu}(t_n)$, leads to

$$\alpha_{\dot{\boldsymbol{\nu}}} = \tfrac{1}{2} \tag{7.29}$$

Similarly, we can find the algorithmic time level of $\nabla\mathscr{U}(\mathbf{q}(t^*_f))$ where $t^*_f := t_n + \alpha_f\Delta t$ for $\alpha_f \in [0, 1] \subset \mathbb{R}$ easily. Consider

$$\nabla\mathscr{U}(\mathbf{q}(t_{n+\alpha_f})) = \nabla\mathscr{U}(\mathbf{q}_{n+\beta_0}) + \mathcal{O}(\Delta t^2) \tag{7.30}$$

Here, from Eq. (7.24), we have

$$\nabla\mathscr{U}(\mathbf{q}_{n+1/2}) = \nabla\mathscr{U}(\mathbf{q}_{n+\beta_0}) + \mathcal{O}(\Delta t^2) \tag{7.31}$$

or

$$\nabla\mathscr{U}(\mathbf{q}(t_{n+1/2})) = \nabla\mathscr{U}(\mathbf{q}_{n+\beta_0}) + \mathcal{O}(\Delta t^2) \tag{7.32}$$

Therefore, we readily find that the algorithmic time level of $\nabla\mathscr{U}$ is also $1/2$, i.e.,

$$\alpha_{\dot{\boldsymbol{\nu}}} = \alpha_f = \tfrac{1}{2} \tag{7.33}$$

Hence, the algorithmic time level of the balance equation of the algorithm is consistent at time $t^* = t_{n+1/2}$,

$$\mathbf{0} = \mathbf{M}\frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t} + \nabla\mathscr{U}(\mathbf{q}_{n+\beta_0}) + \mathcal{O}(\Delta t^2) = \mathbf{M}\dot{\boldsymbol{\nu}}(t^*) + \nabla\mathscr{U}(\mathbf{q}(t^*)) \tag{7.34}$$

The algorithmic time level of the kinematic constraint is easily found to be 1/2, i.e.,

$$\mathbf{0} = \frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t} - \frac{\boldsymbol{\nu}_{n+1} + \boldsymbol{\nu}_n}{2} + \mathcal{O}(\Delta t^2) = \dot{\mathbf{q}}(t_{n+1/2}) - \boldsymbol{\nu}(t_{n+1/2}) \tag{7.35}$$

(details are omitted). It should be noted that the ***algorithmic time levels of the exact energy conserving algorithm and the implicit midpoint rule coincide with each other***.

Linear/Angular momentum conservation within a time step

Under the assumptions we have made in this section, the total linear momentum and the total angular momentum for the system are given as

$$\mathbf{L}(\boldsymbol{\nu}) = \sum_{i=1}^{N} m_i \boldsymbol{\nu}^i \;\text{ and }\; \mathbf{J}(\mathbf{q}, \boldsymbol{\nu}) = \sum_{i=1}^{N} \mathbf{q}^i \times (m_i \boldsymbol{\nu}^i) \tag{7.36}$$

respectively. The conservation of the total linear momentum within a time step $[t_n, t_{n+1}]$ in the sense of $\mathbf{L}_n = \mathbf{L}_{n+1}$, yields

$$\begin{aligned} \frac{\mathbf{L}_{n+1} - \mathbf{L}_n}{\Delta t} &= \sum_{i=1}^{N} m_i \frac{\boldsymbol{\nu}^i_{n+1} - \boldsymbol{\nu}^i_n}{\Delta t} \\ &= \underbrace{-\sum_{i=1}^{N} \sum_{j \neq i}^{N} \frac{\partial \mathscr{U}_{ij}^{\text{int}}(\mathbf{q}^i_{n+\beta_0})}{\partial \mathbf{q}^i_{n+\beta_0}}}_{\mathbf{0} \text{ for any } \beta_0} - \sum_{i=1}^{N} \frac{\partial \mathscr{U}_i^{\text{c}}(\mathbf{q}^i_{n+\beta_0})}{\partial \mathbf{q}^i_{n+\beta_0}} = \sum_{i=1}^{N} \mathbf{f}_i^{\text{c}}(\mathbf{q}_{n+\beta_0}) \end{aligned} \tag{7.37}$$

where $\mathbf{L}_{n+1} := \mathbf{L}(\boldsymbol{\nu}_{n+1}) \approx \mathbf{L}(t_{n+1})$ and $\mathbf{L}_n := \mathbf{L}(\boldsymbol{\nu}_n) \approx \mathbf{L}(t_n)$. The summation of the internal force vanishes for any β_0 due to Newton's third law; hence, the total linear momentum is exactly conserved for any $\beta_0 \in (0, 1)$ in the absence of the total external conservative force.

The conservation of the total angular momentum within a time step $[t_n, t_{n+1}]$

in the sense of $\mathbf{J}_n = \mathbf{J}_{n+1}$ yields

$$\begin{aligned}
\frac{\mathbf{J}_{n+1} - \mathbf{J}_n}{\Delta t} &= \frac{1}{\Delta t}\left[\sum_{i=1}^{N} \mathbf{q}^i_{n+1} \times (m_i \boldsymbol{\nu}^i_{n+1}) - \sum_{i=1}^{N} \mathbf{q}^i_n \times (m_i \boldsymbol{\nu}^i_n)\right] \\
&= \sum_{i=1}^{N} \frac{\mathbf{q}^i_{n+1} + \mathbf{q}^i_n}{2} \times \left(m_i \frac{\boldsymbol{\nu}^i_{n+1} - \boldsymbol{\nu}^i_n}{\Delta t}\right) \\
&\quad + \sum_{i=1}^{N} \frac{\mathbf{q}^i_{n+1} - \mathbf{q}^i_n}{\Delta t} \times \left(m_i \frac{\boldsymbol{\nu}^i_{n+1} + \boldsymbol{\nu}^i_n}{2}\right) \\
&= \sum_{i=1}^{N} \frac{\mathbf{q}^i_{n+1} + \mathbf{q}^i_n}{2} \times \left(-\frac{\partial \mathscr{U}(\mathbf{q}_{n+\beta_0})}{\partial \mathbf{q}^i_{n+\beta_0}}\right) \\
&\quad + \sum_{i=1}^{N} \underbrace{\frac{\boldsymbol{\nu}^i_{n+1} + \boldsymbol{\nu}^i_n}{2} \times \left(m_i \frac{\boldsymbol{\nu}^i_{n+1} + \boldsymbol{\nu}^i_n}{2}\right)}_{\mathbf{0}} \\
&= -\sum_{i=1}^{N} \mathbf{q}^i_{n+1/2} \times \left(\frac{\partial \mathscr{U}(\mathbf{q}_{n+\beta_0})}{\partial \mathbf{q}^i_{n+\beta_0}}\right)
\end{aligned} \tag{7.38}$$

where $\mathbf{J}_{n+1} := \mathbf{J}(\mathbf{q}_{n+1}, \boldsymbol{\nu}_{n+1}) \approx \mathbf{J}(t_{n+1})$ and $\mathbf{J}_n := \mathbf{J}(\mathbf{q}_n, \boldsymbol{\nu}_n) \approx \mathbf{J}(t_n)$. Consider the $SO(3)$-invariance of the total potential energy in the holonomic-scleronomic system. The left rotationally invariance of the total potential energy under the action of the rotation group is

$$\mathscr{U}(\mathbf{q}_{n+\beta_0}) = \mathscr{U}(\mathbf{q}_n, \mathbf{q}_{n+1}, \beta_0) \equiv \mathscr{U}(\mathbf{q}'_n, \mathbf{q}'_{n+1}, \beta_0) \tag{7.39}$$

where

$$\begin{aligned}
\mathbf{q}'_n &= \mathbf{Q} \circ \mathbf{q}_n := (\mathbf{Q}\mathbf{q}^1_n, \cdots, \mathbf{Q}\mathbf{q}^N_n) \in Q \equiv \mathbb{R}^{3N} \\
\mathbf{q}'_{n+1} &= \mathbf{Q} \circ \mathbf{q}_{n+1} := (\mathbf{Q}\mathbf{q}^1_{n+1}, \cdots, \mathbf{Q}\mathbf{q}^N_{n+1}) \in Q \equiv \mathbb{R}^{3N}
\end{aligned} \tag{7.40}$$

with the rotational transformation matrix $\mathbf{Q} = \exp[\epsilon\hat{\boldsymbol{\xi}}] \in SO(3)$. Note $\hat{\boldsymbol{\xi}} \in so(3)$.

The infinitesimally transformed total potential energy is expanded about $\epsilon = 0$ as

$$\begin{aligned}
\phi(\epsilon) &= \mathscr{U}(\mathbf{q}'_n, \mathbf{q}'_{n+1}, \beta_0) \\
&= \mathscr{U}(\exp[\epsilon\hat{\boldsymbol{\xi}}] \circ \mathbf{q}_n, \exp[\epsilon\hat{\boldsymbol{\xi}}] \circ \mathbf{q}_{n+1}, \beta_0) =: \mathscr{U}_\epsilon(\mathbf{q}_n, \mathbf{q}_{n+1}, \beta_0) \\
&= \mathscr{U}(\mathbf{q}_n, \mathbf{q}_{n+1}, \beta_0) + \epsilon \left.\frac{\partial}{\partial\epsilon}\right|_{\epsilon=0} \mathscr{U}_\epsilon(\mathbf{q}_n, \mathbf{q}_{n+1}, \beta_0) \\
&\quad + \frac{\epsilon^2}{2} \left.\frac{\partial^2}{\partial\epsilon^2}\right|_{\epsilon=0} \mathscr{U}_\epsilon(\mathbf{q}_n, \mathbf{q}_{n+1}, \beta_0) + \cdots \\
&= \mathscr{U}(\mathbf{q}_n, \mathbf{q}_{n+1}, \beta_0) + \epsilon\eta(\mathbf{q}_n, \mathbf{q}_{n+1}, \beta_0) + \mathcal{O}(\epsilon^2)
\end{aligned} \tag{7.41}$$

where the infinitesimal of the total energy is defined as

$$\eta(\mathbf{q}_n, \mathbf{q}_{n+1}, \beta_0) = \left.\frac{\partial}{\partial\epsilon}\right|_{\epsilon=0} \mathscr{U}_\epsilon(\mathbf{q}_n, \mathbf{q}_{n+1}, \beta_0) \tag{7.42}$$

The rotational invariance of the total potential energy holds if and only if

$$\begin{aligned}
0 = \eta &= \left.\frac{\partial}{\partial\epsilon}\right|_{\epsilon=0} \mathscr{U}_\epsilon(\mathbf{q}_n, \mathbf{q}_{n+1}, \beta_0) \\
&= (1-\beta_0)\frac{\partial\mathscr{U}(\mathbf{q}_{n+\beta_0})}{\partial\mathbf{q}_{n+\beta_0}} \cdot \left(\hat{\boldsymbol{\xi}}\mathbf{q}_n\right) + \beta_0\frac{\partial\mathscr{U}(\mathbf{q}_{n+\beta_0})}{\partial\mathbf{q}_{n+\beta_0}} \cdot \left(\hat{\boldsymbol{\xi}}\mathbf{q}_{n+1}\right) \\
&= (1-\beta_0)\frac{\partial\mathscr{U}(\mathbf{q}_{n+\beta_0})}{\partial\mathbf{q}_{n+\beta_0}} \cdot (\boldsymbol{\xi} \times \mathbf{q}) + \beta_0\frac{\partial\mathscr{U}(\mathbf{q}_{n+\beta_0})}{\partial\mathbf{q}_{n+\beta_0}} \cdot (\boldsymbol{\xi} \times \mathbf{q}) \\
&= \boldsymbol{\xi} \cdot \left[(1-\beta_0)\mathbf{q}_n \times \frac{\partial\mathscr{U}(\mathbf{q}_{n+\beta_0})}{\partial\mathbf{q}_{n+\beta_0}} + \beta_0\mathbf{q}_{n+1} \times \frac{\partial\mathscr{U}(\mathbf{q}_{n+\beta_0})}{\partial\mathbf{q}_{n+\beta_0}}\right]
\end{aligned} \tag{7.43}$$

Since $\boldsymbol{\xi} \in SO(3)$ is arbitrary, the rotational invariance of the total potential energy is given as

$$(1-\beta_0)\mathbf{q}_n \times \frac{\partial\mathscr{U}(\mathbf{q}_{n+\beta_0})}{\partial\mathbf{q}_{n+\beta_0}} + \beta_0\mathbf{q}_{n+1} \times \frac{\partial\mathscr{U}(\mathbf{q}_{n+\beta_0})}{\partial\mathbf{q}_{n+\beta_0}} = \mathbf{0} \quad \forall t \in \mathbb{I} \tag{7.44}$$

In view of Eq. (7.38) and Eq. (7.44), $SO(3)$ frame invariance shows that we must have $\beta_0 = 1/2$ (symplectic midpoint rule) to exactly conserve the total angular momentum within a time step.

Summarizing, the energy conserving implicit algorithm with $\beta-$iteration may be summarized as follows:

Algorithm 19

Two-Field Form Energy Conserving Algorithm with $\beta-$iteration

Maps $TQ \to TQ$ given by $(\mathbf{q}_n, \boldsymbol{\nu}_n) \mapsto (\mathbf{q}_{n+1}, \boldsymbol{\nu}_{n+1})$ which satisfies

$$\mathbf{M}\frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t} + \nabla \mathscr{U}(\mathbf{q}_{n+\beta_0}) = \mathbf{0} \tag{7.45a}$$

$$\frac{\boldsymbol{\nu}_{n+1} + \boldsymbol{\nu}_n}{2} = \frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t} \tag{7.45b}$$

and there exists some $\beta_0 \in (0,1) \subset \mathbb{R}$ such that

$$\mathscr{U}(\mathbf{q}_{n+1}) - \mathscr{U}(\mathbf{q}_n) = [\mathbf{q}_{n+1} - \mathbf{q}_n] \cdot \nabla \mathscr{U}(\mathbf{q}_{n+\beta_0}) \tag{7.45c}$$

Remark 19 (Algorithm 19)

1. *The pioneering work of constructing the energy-momentum conserving algorithm by means of the finite difference on the derivative of the potential energy has been done by Greenspan [49] and Labudde and Greenspan [50,51].*

2. *Algorithm 19 is an unconditionally stable algorithm in the sense of exact energy conserving within a time step. Recall that the total energy defines the Lyapunov function of the dynamical system.*

3. *The total linear momentum is exactly conserved for any β_0; however, the total angular momentum is exactly conserved only if $\beta_0 = 1/2$ (symplectic midpoint rule). Note symplecticity is also preserved for $\beta_0 = 1/2$ (details are omitted.)*

4. *Algorithm 19 is second-order time accurate for any $\beta_0 \in (0,1)$ which satisfies Eq. (7.45c).*

5. *Based upon the time level consistency theorem (see Theorem 5) for algorithm designs, we can naturally introduce the dissipative and time-dependent external force terms, i.e., $\mathbf{f}^{\text{diss}}(\mathbf{q}, \boldsymbol{\nu})$ and $\mathbf{f}(t)$. Therefore, the balance equation may be modified as*

$$\mathbf{M}\frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t} + \nabla \mathscr{U}(\mathbf{q}_{n+\beta_0}) = \mathbf{f}^{\text{diss}}(\mathbf{q}_{n+1/2}, \boldsymbol{\nu}_{n+1/2}) + \mathbf{f}(t_{n+1/2}) \tag{7.46}$$

In the special case where the dissipative force is given as $\mathbf{f}^{\text{diss}}(\boldsymbol{\nu}) = -\mathbf{C}\boldsymbol{\nu}$, *Eq.* (7.46) *becomes*

$$\mathbf{M}\frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t} + \mathbf{C}\boldsymbol{\nu}_{n+1/2} + \nabla\mathscr{U}(\mathbf{q}_{n+\beta_0}) = \mathbf{f}(t_{n+1/2}) \tag{7.47}$$

6. *When* $\mathbf{q}_{n+1} \approx \mathbf{q}_n$, *Algorithm 19 has a singularity issue; however, the algorithm is well-defined at the limit situation* ($\mathbf{q}_{n+1} \to \mathbf{q}_n$). *The Taylor series expansion shows*

$$\mathscr{U}(\mathbf{q}_{n+1}) - \mathscr{U}(\mathbf{q}_n) = \Delta\mathbf{q}\cdot\nabla\mathscr{U}(\mathbf{q}_{n+\beta}) + \left[\tfrac{1}{2} - \beta\right]\Delta\mathbf{q}\cdot\nabla^2\mathscr{U}(\mathbf{q}_{n+\beta})\Delta\mathbf{q} + \mathcal{O}(\Delta t^3) \tag{7.48}$$

for $\beta \in [0,1]$, *where* $\Delta\mathbf{q} := \mathbf{q}_{n+1} - \mathbf{q}_n$, *and* $\nabla^2\mathscr{U}$ *is the Hessian matrix; therefore, use* $\beta = 1/2 \cong \beta_0$ *as* $\Delta\mathbf{q} \to \mathbf{0}$ *to avoid the singularity issue:*

$$\begin{aligned} \mathscr{U}(\mathbf{q}_{n+1}) - \mathscr{U}(\mathbf{q}_n) &= \Delta\mathbf{q}\cdot\nabla\mathscr{U}(\mathbf{q}_{n+1/2}) + \mathcal{O}(\Delta t^3) \\ &\cong \Delta\mathbf{q}\cdot\nabla\mathscr{U}(\mathbf{q}_{n+1/2}) \ \text{ as } \ \Delta\mathbf{q} \to \mathbf{0} \end{aligned} \tag{7.49}$$

7. *The system Jacobian matrix of Algorithm 19 is usually nonsymmetric whereas it is always symmetric for the symplectic midpoint rule* ($\beta_0 = 1/2$).

Exact Energy-Momentum Conserving Implicit Algorithms

Although Algorithm 19 seems the most general expression for the design of an energy conserving algorithm, it actually has several disadvantages; that is: (1) it requires an extra iterative loop to compute β_0 at every time step, (2) the mean value theorem guarantees the existence of $\beta_0 \in (0,1)$, but there is a possibility that some (more than one) β_0 exist, which may cause inconsistent solutions, and (3) although $\beta_0 \in (0,1)$ must be found via the mean value theorem to achieve the total energy conservation within a time step, only $\beta_0 = 1/2$ guarantees the total angular momentum conservation within a time step. Issue (3) is very severe to design the energy-momentum conserving algorithm; however, this difficulty may be removed by the following procedure. Set $\beta_0 = 1/2$ in Eq. (7.38); that

is, total angular momentum conservation in $[t_n, t_{n+1}]$ in the absence of the total conservative force on a system yields

$$\begin{aligned}\frac{\mathbf{J}_{n+1}-\mathbf{J}_n}{\Delta t} &= \sum_{i=1}^{N}\frac{\mathbf{q}^i_{n+1}+\mathbf{q}^i_n}{2}\times\left(m_i\frac{\boldsymbol{\nu}^i_{n+1}-\boldsymbol{\nu}^i_n}{\Delta t}\right)\\ &\quad+\sum_{i=1}^{N}\frac{\mathbf{q}^i_{n+1}-\mathbf{q}^i_n}{\Delta t}\times\left(m_i\frac{\boldsymbol{\nu}^i_{n+1}+\boldsymbol{\nu}^i_n}{2}\right)\\ &= \underbrace{\sum_{i=1}^{N}\frac{\mathbf{q}^i_{n+1}+\mathbf{q}^i_n}{2}\times\left(-\frac{\partial\mathscr{U}(\mathbf{q}_{n+1/2})}{\partial\mathbf{q}^i_{n+1/2}}\right)}_{\mathbf{0}}\\ &\quad+\underbrace{\sum_{i=1}^{N}\frac{\boldsymbol{\nu}^i_{n+1}+\boldsymbol{\nu}^i_n}{2}\times\left(m_i\frac{\boldsymbol{\nu}^i_{n+1}+\boldsymbol{\nu}^i_n}{2}\right)}_{\mathbf{0}}=\mathbf{0}\end{aligned}\tag{7.50}$$

In Eq.(7.50), notice that the total angular momentum can be still exactly conserved by introducing scalar algorithmic parameters $\kappa_\gamma = 1+\mathcal{O}(\Delta t)\in\mathbb{R}$ which satisfy $\lim_{\Delta t\to 0}\kappa_\gamma = 1$ $(\gamma = 1,2)$ as

$$\begin{aligned}\frac{\mathbf{J}_{n+1}-\mathbf{J}_n}{\Delta t} &= \underbrace{\sum_{i=1}^{N}\frac{\mathbf{q}^i_{n+1}+\mathbf{q}^i_n}{2}\times\left(-\kappa_2\frac{\partial\mathscr{U}(\mathbf{q}_{n+1/2})}{\partial\mathbf{q}^i_{n+1/2}}\right)}_{\mathbf{0}}\\ &\quad+\underbrace{\sum_{i=1}^{N}\kappa_1\frac{\boldsymbol{\nu}^i_{n+1}+\boldsymbol{\nu}^i_n}{2}\times\left(m_i\frac{\boldsymbol{\nu}^i_{n+1}+\boldsymbol{\nu}^i_n}{2}\right)}_{\mathbf{0}}=\mathbf{0}\end{aligned}\tag{7.51}$$

Hence, Algorithm 19 can now be modified as follows so that total energy and total linear/angular momentum are all conserved in $[t_n, t_{n+1}]$:

$$m_i\frac{\boldsymbol{\nu}^i_{n+1}-\boldsymbol{\nu}^i_n}{\Delta t} = -\kappa_2\left[\sum_{j\neq i}^{N}\frac{\partial\mathscr{U}^{\text{int}}_{ij}(\mathbf{q}^i_{n+1/2},\mathbf{q}^j_{n+1/2})}{\partial\mathbf{q}^i_{n+1/2}}+\frac{\partial\mathscr{U}^{\text{c}}_i(\mathbf{q}^i_{n+1/2})}{\partial\mathbf{q}^i_{n+1/2}}\right]\tag{7.52a}$$

$$\frac{\mathbf{q}^i_{n+1}-\mathbf{q}^i_n}{\Delta t} = \kappa_1\frac{\boldsymbol{\nu}^i_{n+1}+\boldsymbol{\nu}^i_n}{2}\tag{7.52b}$$

with the energy conserving enforcement;

$$\begin{aligned}&\sum_{i=1}^{N}\left(\sum_{j=1>i}^{N}\left[\mathscr{U}_{ij}^{\text{int}}(\mathbf{q}_{n+1}^{i},\mathbf{q}_{n+1}^{j})-\mathscr{U}_{ij}^{\text{int}}(\mathbf{q}_{n}^{i},\mathbf{q}_{n}^{j})\right]+\mathscr{U}_{i}^{\text{c}}(\mathbf{q}_{n+1}^{i})-\mathscr{U}_{i}^{\text{c}}(\mathbf{q}_{n}^{i})\right)\\&=\kappa_2\sum_{i=1}^{N}\left[\mathbf{q}_{n+1}^{i}-\mathbf{q}_{n}^{i}\right]\cdot\left[\sum_{j\neq i}^{N}\frac{\partial\mathscr{U}_{ij}^{\text{int}}(\mathbf{q}_{n+1/2}^{i},\mathbf{q}_{n+1/2}^{j})}{\partial\mathbf{q}_{n+1/2}^{i}}+\frac{\partial\mathscr{U}_{i}^{\text{c}}(\mathbf{q}_{n+1/2}^{i})}{\partial\mathbf{q}_{n+1/2}^{i}}\right]\end{aligned}\tag{7.53}$$

This modified algorithm is a special case of the implicit energy-momentum conserving algorithm proposed in Simo *et. al.* [52]. While they derived the algorithm starting from the angular momentum conservation, notice that we have derived it directly from the conservation of the total energy. The algorithmic internal and conservative forces are defined as

$$\tilde{\mathbf{f}}_{\text{con}\,i}^{\text{int}}=\kappa_2\sum_{j\neq i}^{N}\frac{\partial\mathscr{U}_{ij}^{\text{int}}(\mathbf{q}_{n+1/2}^{i},\mathbf{q}_{n+\beta_0}^{j})}{\partial\mathbf{q}_{n+1/2}^{i}}\;\text{ and }\;\tilde{\mathbf{f}}_{\text{con}\,i}^{\text{c}}=-\kappa_2\frac{\partial\mathscr{U}_{i}^{\text{c}}(\mathbf{q}_{n+1/2}^{i})}{\partial\mathbf{q}_{n+1/2}^{i}}\tag{7.54}$$

respectively, in this framework of the energy-momentum conserving algorithm. The implicit energy-momentum conserving algorithm for a system of N particles are summarized as follows:

Algorithm 20

Two-Field Form Energy-Momentum Conserving Algorithm *Maps* $TQ\rightarrow TQ$ *given by* $(\mathbf{q}_n,\boldsymbol{\nu}_n)\mapsto(\mathbf{q}_{n+1},\boldsymbol{\nu}_{n+1})$ *which satisfies*

$$\mathbf{M}\frac{\boldsymbol{\nu}_{n+1}-\boldsymbol{\nu}_n}{\Delta t}+\kappa_2\nabla\mathscr{U}(\mathbf{q}_{n+1/2})=\mathbf{0}\tag{7.55a}$$

$$\frac{\mathbf{q}_{n+1}-\mathbf{q}_n}{\Delta t}=\kappa_1\frac{\boldsymbol{\nu}_{n+1}+\boldsymbol{\nu}_n}{2}\tag{7.55b}$$

$$\textit{with}\quad\mathscr{U}(\mathbf{q}_{n+1})-\mathscr{U}(\mathbf{q}_n)=[\mathbf{q}_{n+1}-\mathbf{q}_n]\cdot\kappa_2\nabla\mathscr{U}(\mathbf{q}_{n+1/2})\tag{7.55c}$$

Remark 20 (Algorithm 20)

1. *Algorithm 20 is an unconditionally stable algorithm in the sense of the exact conservation of the total energy as the Lyapunov function of the dynamical system within a time step.*

2. *Algorithm 20 is the exact energy-momentum conserving algorithm when Eq.*(7.55c) *is solvable. In the absence of the total external force, the total linear and angular momenta are exactly conserved.*

3. *For the numerical implementation of Algorithm 20, we usually set* $\kappa_1 = 1$ *and solve for* $(\mathbf{q}_{n+1}, \kappa_2) \in Q \times \mathbb{R}$ *in d-form or* $(\boldsymbol{\nu}_{n+1}, \kappa_2) \in T_{\mathbf{q}}Q \times \mathbb{R}$ *in v-form. For the solution procedure, see Simo et. al. [52]. The system Jacobian matrix,* $\mathbf{J}$*, of Algorithm 20 is usually nonsymmetric.*

Exact Energy-Momentum Conserving Algorithm for a Distance-Dependent Internal Potential Energy

Consider the special case where the internal potential energy due to interactions between i^{th} and j^{th} particles is given as a function of the inter-particle distance $r^{ij} := \| \mathbf{q}^{ij} \| := \| \mathbf{q}^j - \mathbf{q}^i \| = \sqrt{(\mathbf{q}^j - \mathbf{q}^i) \cdot (\mathbf{q}^j - \mathbf{q}^i)} \in \mathbb{R}$; that is, $\mathscr{U}_{ij}^{\text{int}}(\mathbf{q}^i, \mathbf{q}^j) = \mathscr{V}_{ij}^{\text{int}}(r^{ij})$. For this particular case, Eq. (7.7) can be written explicitly as

$$\begin{aligned} &\sum_{i=1}^{N} \sum_{j>i}^{N} \left[\mathscr{V}_{ij}^{\text{int}}(r_{n+1}^{ij}) - \mathscr{V}_{ij}^{\text{int}}(r_n^{ij}) \right] \\ &= \sum_{i=1}^{N} \sum_{j \neq i}^{N} \left[\mathbf{q}_{n+1}^i - \mathbf{q}_n^i \right] \cdot \left(\frac{d\mathscr{V}_{ij}^{\text{int}}(r^{ij}(\mathbf{q}_{n+\beta_0}))}{dr^{ij}(\mathbf{q}_{n+\beta_0})} \frac{\mathbf{q}_{n+\beta_0}^i - \mathbf{q}_{n+\beta_0}^j}{r^{ij}(\mathbf{q}_{n+\beta_0})} \right) \end{aligned} \tag{7.56}$$

where we define $r_{n+1}^{ij} = r(\mathbf{q}_{n+1}^i, \mathbf{q}_{n+1}^j) := \| \mathbf{q}_{n+1}^j - \mathbf{q}_{n+1}^i \|$, $r_n^{ij} = r(\mathbf{q}_n^i, \mathbf{q}_n^j) := \| \mathbf{q}_n^j - \mathbf{q}_n^i \|$, and $r^{ij}(\mathbf{q}_{n+\beta_0}) = r(\mathbf{q}_{n+\beta_0}^i, \mathbf{q}_{n+\beta_0}^j) := \| \mathbf{q}_{n+\beta_0}^j - \mathbf{q}_{n+\beta_0}^i \|$. Hence, we obtain

$$m_i \frac{\boldsymbol{\nu}_{n+1}^i - \boldsymbol{\nu}_n^i}{\Delta t} + \mathbf{f}_i^{\text{int}_q} = \mathbf{0} \tag{7.57}$$

where

$$\mathbf{f}_i^{\text{int}_q} := \sum_{j \neq i}^{N} \frac{d\mathscr{V}_{ij}^{\text{int}}(r^{ij}(\mathbf{q}_{n+\beta_0}))}{dr^{ij}(\mathbf{q}_{n+\beta_0})} \frac{\mathbf{q}_{n+\beta_0}^i - \mathbf{q}_{n+\beta_0}^j}{r^{ij}(\mathbf{q}_{n+\beta_0})} \tag{7.58}$$

In this special case, we can show

$$\sum_{i=1}^{N} \sum_{j>i}^{N} \left[\mathscr{V}_{ij}^{\text{int}}(r_{n+1}^{ij}) - \mathscr{V}_{ij}^{\text{int}}(r_n^{ij}) \right] = \sum_{i=1}^{N} \left[\mathbf{q}_{n+1}^i - \mathbf{q}_n^i \right] \cdot \mathbf{f}_i^{\text{int}_q} = \sum_{i=1}^{N} \left[\mathbf{q}_{n+1}^i - \mathbf{q}_n^i \right] \cdot \mathbf{f}_i^{\text{int}_r} \tag{7.59}$$

where

$$\mathbf{f}_i^{\mathrm{int}_r} := -\sum_{j\neq i}^{N} \frac{\mathscr{V}_{ij\,n+1}^{\mathrm{int}} - \mathscr{V}_{ij\,n}^{\mathrm{int}}}{r_{n+1}^{ij} - r_n^{ij}} \frac{\mathbf{q}_{n+1}^{ij} + \mathbf{q}_n^{ij}}{r_{n+1}^{ij} + r_n^{ij}} \tag{7.60}$$

Note that $\mathbf{f}_i^{\mathrm{int}_q} - \mathbf{f}_i^{\mathrm{int}}(t_{n+1/2}) = \mathcal{O}(\Delta t^2)$ and $\mathbf{f}_i^{\mathrm{int}_r} - \mathbf{f}_i^{\mathrm{int}}(t_{n+1/2}) = \mathcal{O}(\Delta t^2)$, i.e., $\mathbf{f}_i^{\mathrm{int}_r} = \mathbf{f}_i^{\mathrm{int}_q} + \mathcal{O}(\Delta t^2)$.

Linear/Angular momentum conservations within a time step

It can be clearly shown that the conservation of the total linear momentum within a time step can be achieved from

$$\begin{aligned} \frac{\mathbf{L}_{n+1} - \mathbf{L}_n}{\Delta t} &= \sum_{i=1}^{N} m_i \frac{\boldsymbol{\nu}_{n+1}^i - \boldsymbol{\nu}_n^i}{\Delta t} \\ &= \sum_{i=1}^{N}\sum_{j\neq i}^{N} \frac{\mathscr{V}_{ij\,n+1}^{\mathrm{int}} - \mathscr{V}_{ij\,n}^{\mathrm{int}}}{r_{n+1}^{ij} - r_n^{ij}} \frac{\mathbf{q}_{n+1}^{ij} + \mathbf{q}_n^{ij}}{r_{n+1}^{ij} + r_n^{ij}} = \mathbf{0} \end{aligned} \tag{7.61}$$

And the difference of the total angular momentum within a time step yields

$$\begin{aligned} \frac{\mathbf{J}_{n+1} - \mathbf{J}_n}{\Delta t} = &\underbrace{\sum_{i=1}^{N}\sum_{j\neq i}^{N} \frac{\mathbf{q}_{n+1}^i + \mathbf{q}_n^i}{2} \times \frac{\mathscr{V}_{ij\,n+1}^{\mathrm{int}} - \mathscr{V}_{ij\,n}^{\mathrm{int}}}{r_{n+1}^{ij} - r_n^{ij}} \frac{\mathbf{q}_{n+1}^{ij} + \mathbf{q}_n^{ij}}{r_{n+1}^{ij} + r_n^{ij}}}_{\mathbf{0}} \\ &+ \underbrace{\sum_{i=1}^{N} \frac{\boldsymbol{\nu}_{n+1}^i + \boldsymbol{\nu}_n^i}{2} \times \left(m_i \frac{\boldsymbol{\nu}_{n+1}^i + \boldsymbol{\nu}_n^i}{2} \right)}_{\mathbf{0}} = \mathbf{0} \end{aligned} \tag{7.62}$$

Therefore, The algorithm conserves the momenta and total energy of the system within the time step.

Accuracy

To study the order of time accuracy of the exact energy-momentum algorithm, consider a system with single particle moving in a central force for simplicity (accuracy analysis for a general case can be easily extended). Assume that the

position of the particle is given by $\mathbf{q}(t) : \mathbb{I} \to \mathbb{R}^3$. The equation of motion and the kinematic constraint is given by

$$\begin{aligned} m\dot{\boldsymbol{\nu}} &= -D\mathscr{V}^{\text{int}}(r)\frac{\mathbf{q}}{r} \qquad \forall t \in \mathbb{I} \\ \dot{\mathbf{q}} &= \boldsymbol{\nu} \end{aligned} \tag{7.63}$$

where $D\mathscr{V}^{\text{int}}$ denotes the derivative of $\mathscr{V}^{\text{int}}$ with respect to r, $m \in \mathbb{R}$ is its mass, and $r := \| \mathbf{q} \|$ is the distance from the origin. Employing the exact energy-momentum conserving algorithm, the above system of equations can be temporally discretized as

$$\begin{aligned} m\frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t} &= \frac{\mathscr{V}^{\text{int}}_{n+1} - \mathscr{V}^{\text{int}}_n}{r_{n+1} - r_n}\frac{\mathbf{q}_{n+1} + \mathbf{q}_n}{r_{n+1} + r_n} \\ \frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t} &= \frac{\boldsymbol{\nu}_{n+1} + \boldsymbol{\nu}_n}{2} \end{aligned} \tag{7.64}$$

Define the local truncation error by $\boldsymbol{\tau}(\Delta t) = (\boldsymbol{\tau}_1(\Delta t), \boldsymbol{\tau}_2(\Delta t))^T$, where

$$\begin{aligned} \boldsymbol{\tau}_1(\Delta t) &:= \mathbf{M}\left[\boldsymbol{\nu}(t_{n+1}) - \boldsymbol{\nu}(t_n)\right]\left[r(t_{n+1}) + r(t_n)\right] \\ &\quad + \Delta t \bar{P}\left[\mathbf{q}(t_{n+1}) + \mathbf{q}(t_n)\right] \equiv \mathbf{0} \end{aligned} \tag{7.65}$$

$$\boldsymbol{\tau}_2(\Delta t) := \mathbf{q}(t_{n+1}) - \mathbf{q}(t_n) - \frac{\Delta t}{2}\left[\boldsymbol{\nu}(t_{n+1}) + \boldsymbol{\nu}(t_n)\right] \equiv \mathbf{0} \tag{7.66}$$

in which $\bar{P}$ denotes the exact value of $(\mathscr{V}^{\text{int}}_{n+1} - \mathscr{V}^{\text{int}}_n)/(r_{n+1} - r_n)$.

We have $\boldsymbol{\tau}_2(\Delta t) = \mathcal{O}(\Delta t^3)$ and $\| \boldsymbol{\tau}_2(\Delta t) \| = \mathcal{O}(\Delta t^3)$ from Eq. (7.26).

Algorithm 21

Two-field Form Exact Energy-Momentum Conserving Algorithm (Special Case)

Maps TQ → TQ given by $(\mathbf{q}_n, \boldsymbol{\nu}_n) \mapsto (\mathbf{q}_{n+1}, \boldsymbol{\nu}_{n+1})$ *which satisfies*

$$\begin{aligned} m_i\frac{\boldsymbol{\nu}^i_{n+1} - \boldsymbol{\nu}^i_n}{\Delta t} - \sum_{j \neq i}^{N} \frac{\mathscr{V}^{\text{int}}_{ij\,n+1} - \mathscr{V}^{\text{int}}_{ij\,n}}{r^{ij}_{n+1} - r^{ij}_n}\frac{\mathbf{q}^{ij}_{n+1} + \mathbf{q}^{ij}_n}{r^{ij}_{n+1} + r^{ij}_n} &= \mathbf{0} \\ \frac{\mathbf{q}^i_{n+1} - \mathbf{q}^i_n}{\Delta t} &= \frac{\boldsymbol{\nu}^i_{n+1} + \boldsymbol{\nu}^i_n}{2} \end{aligned}$$

Initial conditions:

$$\begin{aligned} \mathbf{q}(t_0) &= \mathbf{q}_0 \\ \boldsymbol{\nu}(t_0) &= \boldsymbol{\nu}_0 \end{aligned}$$

Remark 21 (Algorithm 21)

1. *Algorithm 21 is second-order time accurate, and the total linear and angular momenta and total energy in* $[t_n, t_{n+1}]$ *are exactly conserved.*

2. *Assuming that the algorithm is well-defined, we employ*

$$\left.\frac{d\mathscr{V}_{ij}^{\text{int}}}{dr^{ij}}\right|_{r_{n+1/2}^{ij}} \quad \textit{instead of} \quad \left.\frac{d\mathscr{V}_{ij}^{\text{int}}}{dr^{ij}}\right|_{r_{n+\phi_0}^{ij}} = \frac{\mathscr{V}_{ij\,n+1}^{\text{int}} - \mathscr{V}_{ij\,n}^{\text{int}}}{r_{n+1}^{ij} - r_n^{ij}} \tag{7.67}$$

for the limit case $(r_{n+1}^{ij} - r_n^{ij} \to 0)$.

7.1.2 Single-field Form: Extensions to Nonlinear Dynamical Systems via Normalized Time Weighted Residual Methodology - Energy-Momentum I-GSSSS Framework - Option III

Algorithmic Time Levels of the Implicit GSSSS Family of Algorithms and the Discrete Energy Conservation

In contrast to Options I and II described in the previous chapter, here we show Option III for developing energy-momentum conserving/dissipative algorithms and designs. First, we show the relation between the exact energy and energy-momentum conserving schemes we have just derived and the implicit GSSSS algorithms in terms of the algorithmic time levels of these schemes. This is possible by studying the conditions of the algorithmic parameters of the implicit GSSSS algorithms that satisfy the exact discrete energy conservation within the time step. In the single-field form, we can obtain the following equation via the mean value theorem on the mechanical energy $\mathcal{E}(\mathbf{q}, \dot{\mathbf{q}}) = \mathcal{T}(\dot{\mathbf{q}}) + \mathcal{U}(\mathbf{q}) : TQ \to \mathbb{R}$ in time,

$$0 = \frac{\mathcal{E}_{n+1} - \mathcal{E}_n}{\Delta t} = \dot{\mathbf{q}}_{n+1/2} \cdot \mathbf{M} \frac{\dot{\mathbf{q}}_{n+1} - \dot{\mathbf{q}}_n}{\Delta t} + \nabla \mathcal{U}(\mathbf{q}_{n+\beta_0}) \cdot \frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t} \tag{7.68}$$

where $\beta_0 \in (0, 1)$ satisfies

$$[\mathbf{q}_{n+1} - \mathbf{q}_n] \cdot \nabla \mathcal{U}(\mathbf{q}_{n+\beta_0}) = \mathcal{U}_{n+1} - \mathcal{U}_n \tag{7.69}$$

From the updates of the GSSSS family of Algorithms in the single-field form, we have

$$\frac{\dot{\mathbf{q}}_{n+1}-\dot{\mathbf{q}}_n}{\Delta t} = \tilde{\mathbf{a}}+(\lambda_5-W_1\Lambda_6)\Delta\mathbf{a} \tag{7.70}$$

$$\frac{\mathbf{q}_{n+1}-\mathbf{q}_n}{\Delta t} = \dot{\mathbf{q}}_{n+1/2}+\Delta t(\lambda_3-\frac{\lambda_5}{2})\Delta\mathbf{a} \tag{7.71}$$

where $\tilde{\mathbf{a}}=(1-W_1\Lambda_6)\ddot{\mathbf{q}}_n+W_1\Lambda_6\ddot{\mathbf{q}}_{n+1}=\ddot{\mathbf{q}}(t_{n+W_1\Lambda_6})+\mathcal{O}(\Delta t^2)$ denotes the algorithmic acceleration vector. Hence, Eq. (7.68) yields

$$\begin{aligned}
0 &= \dot{\mathbf{q}}_{n+1/2}\cdot\mathbf{M}\left[\tilde{\mathbf{a}}+(\lambda_5-W_1\Lambda_6)\Delta\mathbf{a}\right] \\
&\quad +\nabla\mathcal{U}(\mathbf{q}_{n+\beta_0})\cdot\left[\dot{\mathbf{q}}_{n+1/2}+\Delta t(\lambda_3-\frac{\lambda_5}{2})\Delta\mathbf{a}\right] \\
&= \dot{\mathbf{q}}_{n+1/2}\cdot[\mathbf{M}\tilde{\mathbf{a}}+\nabla\mathcal{U}(\mathbf{q}_{n+\beta_0})] \\
&\quad +\Delta\mathbf{a}\cdot\left[(\lambda_5-W_1\Lambda_6)\mathbf{M}\dot{\mathbf{q}}_{n+1/2}+\Delta t(\lambda_3-\frac{\lambda_5}{2})\nabla\mathcal{U}(\mathbf{q}_{n+\beta_0})\right] \\
&= \dot{\mathbf{q}}_{n+1/2}\cdot[\nabla\mathcal{U}(\mathbf{q}_{n+\beta_0})-\nabla\mathcal{U}(\tilde{\mathbf{q}})] \\
&\quad +\Delta\mathbf{a}\cdot\left[(\lambda_5-W_1\Lambda_6)\mathbf{M}\dot{\mathbf{q}}_{n+1/2}+\Delta t(\lambda_3-\frac{\lambda_5}{2})\nabla\mathcal{U}(\mathbf{q}_{n+\beta_0})\right]
\end{aligned} \tag{7.72}$$

where we have used the relations, $\mathbf{M}\tilde{\mathbf{a}}+\nabla\mathcal{U}(\tilde{\mathbf{q}})=\mathbf{0}$ in the conservative system in which $\tilde{\mathbf{q}}$ denotes the algorithmic configuration vector, and

$$\dot{\mathbf{q}}_{n+1/2}\cdot\mathbf{M}\tilde{\mathbf{a}}=\tilde{\mathbf{a}}\cdot\mathbf{M}^T\dot{\mathbf{q}}_{n+1/2}=\tilde{\mathbf{a}}\cdot\mathbf{M}\dot{\mathbf{q}}_{n+1/2} \tag{7.73}$$

since $\mathbf{M}$ is symmetric. The necessary spectral condition for the satisfaction of the exact conservation of the discrete mechanical energy in $[t_n,t_{n+1}]$ in the sense of $\mathcal{E}_{n+1}=\mathcal{E}_n$ is that the maximum principle root is unity, i.e., $\rho_\infty^{\max}=1$. The ***U0-based family*** of algorithms with $\rho_\infty^{\max}=1$ and $\lambda_i=\Lambda_i$ (for $i=1,2,\cdots,5$) gives

$$\begin{aligned}
\tilde{\mathbf{q}}=\mathbf{q}_{n+W_1} &= (1-\frac{1}{1+\rho_\infty^{\mathrm{s}}})\mathbf{q}_n+\frac{1}{1+\rho_\infty^{\mathrm{s}}}\mathbf{q}_{n+1} \\
\Lambda_6 &= \frac{3+\rho_\infty^{\min}+\rho_\infty^{\mathrm{s}}-\rho_\infty^{\min}\rho_\infty^{\mathrm{s}}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\mathrm{s}})} \\
\lambda_5-W_1\Lambda_6 &= \frac{\rho_\infty^{\mathrm{s}}-1}{2(\rho_\infty^{\mathrm{s}}+1)} \\
\lambda_3-\frac{\lambda_5}{2} &= 0
\end{aligned} \tag{7.74}$$

Therefore, Eq. (7.72) can be written as

$$0 = \dot{\mathbf{q}}_{n+1/2} \cdot \left[\nabla \mathcal{U}(\mathbf{q}_{n+\beta_0}) - \nabla \mathcal{U}(\mathbf{q}_{n+W_1}) + \frac{\rho_\infty^{\mathrm{s}} - 1}{2(\rho_\infty^{\mathrm{s}} + 1)} \mathbf{M} \Delta \mathbf{a} \right] \tag{7.75}$$

Since $\dot{\mathbf{q}}_{n+1/2}$ is linearly independent, we can conclude that the following relation must hold

$$\nabla \mathcal{U}(\mathbf{q}_{n+\beta_0}) = \nabla \mathcal{U}(\mathbf{q}_{n+W_1}) - \frac{\rho_\infty^{\mathrm{s}} - 1}{2(\rho_\infty^{\mathrm{s}} + 1)} \mathbf{M} \Delta \mathbf{a} \tag{7.76}$$

so that the discrete mechanical energy within the time step, in the sense of $\mathcal{E}_{n+1} = \mathcal{E}_n$, is exactly conserved in the conservative system. Recalling the relation,

$$\begin{aligned} \partial_i \mathcal{U}(\mathbf{q}_{n+\beta_0}) - \partial_i \mathcal{U}(\mathbf{q}_{n+W_1}) = & \left(\frac{1}{2} - W_1 \right) \partial_i \partial_j \mathcal{U}(\mathbf{q}_n) \Delta q^j \\ & + \left(\frac{1}{6} - \frac{W_1^2}{2} \right) \partial_i \partial_j \partial_k \mathcal{U}(\mathbf{q}_n) \Delta q^j \Delta q^k + \mathcal{O}(\Delta t^3) \end{aligned} \tag{7.77}$$

where $\partial_i \equiv \partial / \partial q^i$ denotes the derivative with respect to q^i, Eq. (7.76) can be also written as

$$\begin{aligned} 0 = & \frac{\rho_\infty^{\mathrm{s}} - 1}{2(\rho_\infty^{\mathrm{s}} + 1)} M_{ij} \Delta a_j + \left(\frac{1}{2} - W_1 \right) \partial_i \partial_j \mathcal{U}(\mathbf{q}_n) \Delta q^j \\ & + \left(\frac{1}{6} - \frac{W_1^2}{2} \right) \partial_i \partial_j \partial_k \mathcal{U}(\mathbf{q}_n) \Delta q^j \Delta q^k + \mathcal{O}(\Delta t^3) \end{aligned} \tag{7.78}$$

where $M_{ij} = [\mathbf{M}]_{ij}$. Substituting $W_1 = 1/(1 + \rho_\infty^{\mathrm{s}})$ into Eq. (7.78), yields

$$\begin{aligned} 0 = & \frac{\rho_\infty^{\mathrm{s}} - 1}{2(\rho_\infty^{\mathrm{s}} + 1)} \left[M_{ij} \Delta a_j + \partial_i \partial_j \mathcal{U}(\mathbf{q}_n) \Delta q^j \right] \\ & + \frac{(\rho_\infty^{\mathrm{s}})^2 + 2\rho_\infty^{\mathrm{s}} - 2}{6(1 + \rho_\infty^{\mathrm{s}})^2} \partial_i \partial_j \partial_k \mathcal{U}(\mathbf{q}_n) \Delta q^j \Delta q^k + \mathcal{O}(\Delta t^3) \end{aligned} \tag{7.79}$$

In the ***linear*** dynamical system where the potential energy is defined as a quadratic function in $\mathbf{q} \in Q$, i.e., $\mathcal{U} = (1/2)\mathbf{q} \cdot \mathbf{K}\mathbf{q}$ where $[\mathbf{K}]_{ij} = \partial_i \partial_j \mathcal{U}$ is symmetric and positive semi-definite, Eq. (7.79) becomes

$$0 = \frac{\rho_\infty^{\mathrm{s}} - 1}{2(\rho_\infty^{\mathrm{s}} + 1)} \left[M_{ij} \Delta a_j + K_{ij} \Delta q^j \right] \tag{7.80}$$

since $\partial_k K_{ij} = 0$. The question is which member(s) in the U0-based family of algorithms can satisfy Eq. (7.80)?

Firstly, in the ***U0-based family*** of algorithms, $\mathbf{M}\Delta\mathbf{a} + \mathbf{K}\Delta\mathbf{q} = \mathbf{0}$ holds if and only if $\Lambda_6 = 1$, i.e., $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$. Hence, the discrete mechanical energy in the time step is exactly conserved in the linear conservative system if and only if $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ (notice that $\rho_\infty^{\mathrm{s}} \in [0, 1]$ remains as a free parameter). However, in the nonlinear conservative system, this is not true. For example, the implicit Newmark algorithm, U0 $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}}) = (1, 1, 0)$, i.e., a particular member of U0 $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}}) = (1, 1, \rho_\infty^{\mathrm{s}})$, is the exact energy (and actually momentum) conserving scheme in the linear dynamical system, but it is not true in the nonlinear case. It is also important to note that the algorithmic time level of the exact energy conserving schemes in the U0 family of algorithms, i.e., U0 $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}}) = (1, 1, \rho_\infty^{\mathrm{s}})$, is $t^* = t_{n+W_1}$ where $W_1 = 1/(1 + \rho_\infty^{\mathrm{s}}) \in [1/2, 1]$.
In the general ***nonlinear*** dynamical system, we see from Eq. (7.79) that

$$\partial_i \mathcal{U}(\mathbf{q}_{n+\beta_0}) - \left[\partial_i \mathcal{U}(\mathbf{q}_{n+W_1}) + \frac{\rho_\infty^{\mathrm{s}} - 1}{2(\rho_\infty^{\mathrm{s}} + 1)} M_{ij} \Delta a_j\right] = \mathcal{O}(\Delta t^2) \tag{7.81}$$

only if we select $\rho_\infty^{\min} = \rho_\infty^{\max} = \rho_\infty^{\mathrm{s}} = 1$, i.e., $t^* = t_{n+1/2}$; otherwise, if we select $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\mathrm{s}} \in [0, 1)$, for example, it becomes only $\mathcal{O}(\Delta t)$, and note that the right-hand side of the above equation vanishes only in the linear system where we always have $\beta_0 = 1/2$. This implies that the algorithmic time levels of the exact energy conserving scheme and the MPR-EPA scheme are the same at $t^* = t_{n+1/2}$ in the nonlinear system. That is, we need to design the algorithmic conservative force vector, $\tilde{\mathbf{f}}^{\mathrm{con}}$, which exactly satisfies

$$[\mathbf{q}_{n+1} - \mathbf{q}_n] \cdot \tilde{\mathbf{f}}^{\mathrm{con}} = \mathcal{U}_{n+1} - \mathcal{U}_n \tag{7.82}$$

keeping the algorithmic time level of $t^* = t_{n+1/2}$, i.e.,

$$\tilde{f}_i^{\mathrm{con}} - f_i^{\mathrm{con}}(t_{n+1/2}) = \mathcal{O}(\Delta t^2) \tag{7.83}$$

and, it should be recovered when we select $\rho_\infty^{\min} = \rho_\infty^{\max} = \rho_\infty^{\mathrm{s}} = 1$ in the U0 family of algorithms. This is in order to construct the GSSSS family of the Energy

conserving and dissipative schemes as an extension of the U0 family of the GSSSS algorithms in the sense of *Option II*.

On the other hand, secondly, the ***V0-based family*** of algorithms with $\rho_\infty^{\max} = 1$ and $\lambda_i = \Lambda_i$ (for $i = 1, 2, \cdots, 5$) gives

$$\begin{aligned} \tilde{\mathbf{q}} = \mathbf{q}_{n+W_1} &= (1 - \frac{1}{1+\rho_\infty^{\min}})\mathbf{q}_n + \frac{1}{1+\rho_\infty^{\min}}\mathbf{q}_{n+1} \\ \Lambda_6 &= \frac{3 + \rho_\infty^{\min} + \rho_\infty^{\mathrm{s}} - \rho_\infty^{\min}\rho_\infty^{\mathrm{s}}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\mathrm{s}})} \\ \lambda_5 - W_1\Lambda_6 &= \frac{\rho_\infty^{\min} - 1}{2(\rho_\infty^{\min}+1)} \\ \lambda_3 - \frac{\lambda_5}{2} &= 0 \text{ (always)} \end{aligned} \tag{7.84}$$

Therefore, Eq. (7.72) can be written as

$$0 = \dot{\mathbf{q}}_{n+1/2} \cdot \left[\nabla\mathcal{U}(\mathbf{q}_{n+\beta_0}) - \nabla\mathcal{U}(\mathbf{q}_{n+W_1}) + \frac{\rho_\infty^{\min} - 1}{2(\rho_\infty^{\min}+1)}\mathbf{M}\Delta\mathbf{a} \right] \tag{7.85}$$

Since $\dot{\mathbf{q}}_{n+1/2}$ is linearly independent, we can conclude that the following relation must hold

$$\nabla\mathcal{U}(\mathbf{q}_{n+\beta_0}) = \nabla\mathcal{U}(\mathbf{q}_{n+W_1}) - \frac{\rho_\infty^{\min} - 1}{2(\rho_\infty^{\min}+1)}\mathbf{M}\Delta\mathbf{a} \tag{7.86}$$

so that the discrete mechanical energy within the time step, in the sense of $\mathcal{E}_{n+1} = \mathcal{E}_n$, is exactly conserved in the conservative system. Substituting Eq. (7.77), Eq. (7.76) becomes

$$\begin{aligned} 0 = &\frac{\rho_\infty^{\min} - 1}{2(\rho_\infty^{\mathrm{s}}+1)} M_{ij}\Delta a_j + \left(\frac{1}{2} - W_1\right)\partial_i\partial_j\mathcal{U}(\mathbf{q}_n)\Delta q^j \\ &+ \left(\frac{1}{6} - \frac{W_1^2}{2}\right)\partial_i\partial_j\partial_k\mathcal{U}(\mathbf{q}_n)\Delta q^j\Delta q^k + \mathcal{O}(\Delta t^3) \end{aligned} \tag{7.87}$$

Substituting $W_1 = 1/(1+\rho_\infty^{\min})$ into Eq. (7.78), we get

$$\begin{aligned} 0 = &\frac{\rho_\infty^{\min} - 1}{2(\rho_\infty^{\min}+1)} \left[M_{ij}\Delta a_j + \partial_i\partial_j\mathcal{U}(\mathbf{q}_n)\Delta q^j \right] \\ &+ \frac{(\rho_\infty^{\min})^2 + 2\rho_\infty^{\min} - 2}{6(1+\rho_\infty^{\min})^2}\partial_i\partial_j\partial_k\mathcal{U}(\mathbf{q}_n)\Delta q^j\Delta q^k + \mathcal{O}(\Delta t^3) \end{aligned} \tag{7.88}$$

In the ***linear*** dynamical system, Eq. (7.88) becomes

$$0 = \frac{\rho_\infty^{\min} - 1}{2(\rho_\infty^{\min} + 1)} \left[M_{ij} \Delta a_j + K_{ij} \Delta q^j \right] \tag{7.89}$$

In the V0-family of algorithms, $\mathbf{M}\Delta\mathbf{a} + \mathbf{K}\Delta\mathbf{q} = \mathbf{0}$ holds if and only if $\Lambda_6 = 1$ which gives $\rho_\infty^{\min} = \rho_\infty^{\max} = \rho_\infty^{\mathrm{s}} = 1$. Obviously, this is a too strict condition for the exact energy conservation within the time step. The necessary and sufficient spectral condition for the exact energy conservation in the linear dynamical system within the V0 family of algorithms is $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\mathrm{s}} \in [0, 1]$. The algorithmic time level of the family of the schemes is $t^* = t_{n+1/2}$. When selecting $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$, we can reduce the right-hand side of Eq. (7.86) to $\nabla\mathcal{U}(\mathbf{q}_{n+1/2})$. Since $\beta_0 = 1/2$ is true only for the linear system, Eq. (7.86) can never hold in the general nonlinear dynamical system. If we select $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$, Eq. (7.88) implies

$$\partial_i \mathcal{U}(\mathbf{q}_{n+\beta_0}) - \partial_i \mathcal{U}(\mathbf{q}_{n+W_1}) = \mathcal{O}(\Delta t^2) \tag{7.90}$$

for any $\rho_\infty^{\mathrm{s}} \in [0, 1]$. Therefore, we must design an algorithmic conservative force vector, $\tilde{\mathbf{f}}^{\mathrm{con}}$, which exactly satisfies

$$[\mathbf{q}_{n+1} - \mathbf{q}_n] \cdot \tilde{\mathbf{f}}^{\mathrm{con}} = \mathcal{U}_{n+1} - \mathcal{U}_n \tag{7.91}$$

keeping the algorithmic time level of $t^* = t_{n+1/2}$, i.e.,

$$\tilde{f}_i^{\mathrm{con}} - f_i^{\mathrm{con}}(t_{n+1/2}) = \mathcal{O}(\Delta t^2) \tag{7.92}$$

We hence modify the V0-based family of algorithms such that we can recover $\tilde{\mathbf{f}}^{\mathrm{con}}$ when selecting $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ with $\rho_\infty^{\mathrm{s}} \in [0, 1]$ in order to include the exact energy conserving schemes within the framework of the GSSSS family of algorithms as an extension of the V0 family of the GSSSS algorithms in the sense of *Option II*.

Derivation of a New Framework of Energy-Momentum Conserving and Dissipative Implicit GSSSS Algorithms (Option III and Modified Option III)

Since we have seen that the algorithmic time level of the exact energy-momentum conserving schemes and the implicit GSSSS family of algorithms are the same at the midpoint, i.e., $W_1 = 1/2$, we now derive new families of the implicit GSSSS algorithms, ***Option III*** and ***Modified Option III***, so that the energy-momentum conserving schemes are naturally and completely included within the algorithmic framework. The numerically dissipative algorithms which are naturally obtained from the new implicit family of the GSSSS algorithms are based on the exact energy-momentum conserving schemes.

Consider the dynamical system where the autonomous total energy of the system is given as the summation of the kinetic energy $\mathscr{T}$, internal potential energy given as a function of the inter-particle distance $\mathscr{V}$, and the external potential energy $\mathscr{U}^{\mathrm{c}}$, i.e.,

$$\begin{aligned} \mathscr{E} &= \mathscr{T} + \mathscr{U} = \mathscr{T} + \mathscr{V} + \mathscr{U}^{\mathrm{c}} \\ &= \frac{1}{2}\sum_{i=1}^{N} m_i \dot{\mathbf{q}}^i \cdot \dot{\mathbf{q}}^i + \sum_{i=1}^{N}\sum_{j>i}^{N} \mathscr{V}_{ij}(r^{ij}) + \sum_{i=1}^{N} \mathscr{U}_i^{\mathrm{c}}(\mathbf{q}^i) \end{aligned} \tag{7.93}$$

where $r^{ij} := \|\mathbf{q}^{ij}\| = \|\mathbf{q}^i - \mathbf{q}^j\|$ denotes the inter-particle distance between particles i and j for $i, j = 1, 2, \cdots, N$. The principle of balance of mechanical energy with the initial conditions leads to the initial-value problem constituted of the following balance equation and the initial conditions:

$$\boxed{\begin{array}{l} \textbf{\textit{Balance Equation:}} \\ m_i \ddot{\mathbf{q}}^i(t) - \displaystyle\sum_{j\neq i}^{N} \mathscr{V}'_{ij}(r^{ij}) \frac{\mathbf{q}^{ij}}{r^{ij}} = \mathbf{f}_i^{\mathrm{ext}}(\mathbf{q}, \dot{\mathbf{q}}, t) \;\forall t \in \mathbb{I} \\ \textbf{\textit{Initial conditions:}} \\ \mathbf{q}^i(t_0) = \mathbf{q}_0^i \text{ and } \dot{\mathbf{q}}^i(t_0) = \dot{\mathbf{q}}_0^i \end{array}} \tag{7.94}$$

for $i, j = 1, 2, \cdots, N$ (note: $\mathscr{V}'_{ij} = d\mathscr{V}_{ij}/dr^{ij}$). From the balance equation above, we get

$$m_i \ddot{\mathbf{q}}^i(t) - \sum_{j\neq i}^{N} \mathscr{V}'_{ij}(r^{ij}) \frac{\mathbf{q}^{ij}}{r^{ij}} - \mathbf{f}_i^{\mathrm{ext}}(\mathbf{q}, \dot{\mathbf{q}}, t) = \mathbf{0} \;\; \forall t \in \mathbb{I} \tag{7.95}$$

Assume that the (total) external force vector, $\mathbf{f}_i^{\text{ext}}(\mathbf{q},\dot{\mathbf{q}},t) : TQ \times \mathbb{I} \to \mathbb{R}^{3N}$, is given as the summation of $\mathbf{f}^{\text{e}}(\mathbf{q},\dot{\mathbf{q}}) : TQ \to \mathbb{R}^{3N}$ and the time-dependent external force vector $\mathbf{f}(t) : \mathbb{I} \to \mathbb{R}^{3N}$ as $\mathbf{f}^{\text{ext}}(\mathbf{q},\dot{\mathbf{q}},t) = \mathbf{f}^{\text{e}}(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{f}(t)$. In *Option III*, we discretize $\mathbf{q}(t) : \mathbb{I} \to Q \equiv \mathbb{R}^{3N}$ and the inter-particle distance independently to discretize the internal force vector, employing the normalized time weighted residual methodology. The asymptotic series expansions of $\mathbf{q}$, $\dot{\mathbf{q}}$, $\ddot{\mathbf{q}}$, and r^{ij} about time $t = t_{n+\alpha} := (1-\alpha)t_n + \alpha t_{n+1} \in [t_n, t_{n+1}]$ for $\alpha \in [0,1]$ yield

$$\begin{aligned}
\mathbf{q}(t_{n+\alpha}) &\cong \mathbf{q}(t_n) + \Lambda_1\dot{\mathbf{q}}(t_n)[t_{n+\alpha}-t_n] + \Lambda_2\ddot{\mathbf{q}}(t_n)[t_{n+\alpha}-t_n]^2 \\
&\quad + \Lambda_3\frac{\ddot{\mathbf{q}}(t_{n+1})-\ddot{\mathbf{q}}(t_n)}{\Delta t}[t_{n+\alpha}-t_n]^3 =: \hat{\mathbf{q}} \qquad (7.96)\\
\dot{\mathbf{q}}(t_{n+\alpha}) &\cong \dot{\mathbf{q}}(t_n) + \Lambda_4\ddot{\mathbf{q}}(t_n)[t_{n+\alpha}-t_n] \\
&\quad + \Lambda_5\frac{\ddot{\mathbf{q}}(t_{n+1})-\ddot{\mathbf{q}}(t_n)}{\Delta t}[t_{n+\alpha}-t_n]^2 =: \hat{\mathbf{v}} \qquad (7.97)\\
\ddot{\mathbf{q}}(t_{n+\alpha}) &\cong \ddot{\mathbf{q}}(t_n) + \Lambda_6\frac{\ddot{\mathbf{q}}(t_{n+1})-\ddot{\mathbf{q}}(t_n)}{\Delta t}[t_{n+\alpha}-t_n] =: \hat{\mathbf{q}} \qquad (7.98)
\end{aligned}$$

and

$$\begin{aligned}
r^{ij}(t_{n+\alpha}) \cong & r^{ij}(t_n) + \Theta_1\dot{r}^{ij}(t_n)[t_{n+\alpha}-t_n] + \Theta_2\ddot{r}^{ij}(t_n)[t_{n+\alpha}-t_n]^2 \\
& + \Theta_3\frac{\ddot{r}^{ij}(t_{n+1})-\ddot{r}^{ij}(t_n)}{\Delta t}[t_{n+\alpha}-t_n]^3 =: \hat{r}^{ij}
\end{aligned} \qquad (7.99)$$

respectively with new scalar algorithmic parameters, $\Theta_k \in \mathbb{R}$ for $k = 1,2,3$. For the time-dependent external force vector, we linearly approximate within a time step $[t_n, t_{n+1}]$ as

$$\mathbf{f}(t_{n+\alpha}) \cong \mathbf{f}(t_n) + \frac{\mathbf{f}(t_{n+1})-\mathbf{f}(t_n)}{\Delta t}[t_{n+\alpha}-t_n] =: \hat{\mathbf{f}} \qquad (7.100)$$

Substituting Eq. (7.96) - Eq. (7.100) into Eq. (7.95), the residual, $\mathbf{r}_{\text{III}}$, is defined as

$$\mathbf{r}_{\text{III}} := m_i\hat{\mathbf{a}}^i - \sum_{j\neq i}^{N}\mathscr{V}'_{ij}(\hat{r}^{ij})\frac{\hat{\mathbf{q}}^{ij}}{\hat{r}^{ij}} - \mathbf{f}_i^{\text{e}}(\hat{\mathbf{q}},\hat{\mathbf{v}}) - \hat{\mathbf{f}} \qquad (7.101)$$

Following the normalized time weighted residual methodology, it yields

$$\begin{aligned} m_i \frac{\int_0^{\Delta t} W \hat{\mathbf{a}}^i d\tau}{\int_0^{\Delta t} W d\tau} - \sum_{j \neq i}^{N} \mathscr{V}'_{ij}\left(\frac{\int_0^{\Delta t} W \hat{r}^{ij} d\tau}{\int_0^{\Delta t} W d\tau}\right) \left[\frac{\int_0^{\Delta t} W \hat{\mathbf{q}}^{ij} d\tau}{\int_0^{\Delta t} W d\tau} \Big/ \frac{\int_0^{\Delta t} W \hat{r}^{ij} d\tau}{\int_0^{\Delta t} W d\tau} \right] \\ - \mathbf{f}_i^{\mathrm{e}}\left(\frac{\int_0^{\Delta t} W \hat{\mathbf{q}} d\tau}{\int_0^{\Delta t} W d\tau}, \frac{\int_0^{\Delta t} W \hat{\mathbf{v}} d\tau}{\int_0^{\Delta t} W d\tau}\right) - \frac{\int_0^{\Delta t} W \hat{\mathbf{f}}_i d\tau}{\int_0^{\Delta t} W d\tau} \cong \mathbf{0} \end{aligned} \tag{7.102}$$

where $\tau := t_{n+\alpha} - t_n = \alpha \Delta t$ or

$$m_i \tilde{\mathbf{a}}^i - \sum_{j \neq i}^{N} \mathscr{V}'_{ij}(\tilde{r}^{ij}) \frac{\tilde{\mathbf{q}}^{ij}}{\tilde{r}^{ij}} = \mathbf{f}_i^{\mathrm{e}}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}) + \tilde{\mathbf{f}}_i \tag{7.103}$$

where

$$\tilde{\mathbf{q}} = \mathbf{q}_n + W_1 \Lambda_1 \dot{\mathbf{q}}_n \Delta t + W_2 \Lambda_2 \ddot{\mathbf{q}}_n \Delta t^2 + W_3 \Lambda_3 \Delta \mathbf{a} \Delta t^2 \tag{7.104}$$

$$\tilde{\mathbf{v}} = \dot{\mathbf{q}}_n + W_1 \Lambda_4 \ddot{\mathbf{q}}_n \Delta t + W_2 \Lambda_5 \Delta \mathbf{a} \Delta t \tag{7.105}$$

$$\tilde{\mathbf{a}} = \ddot{\mathbf{q}}_n + W_1 \Lambda_6 \Delta \mathbf{a} \tag{7.106}$$

$$\tilde{\mathbf{f}} = \mathbf{f}_n + W_1 (\mathbf{f}_{n+1} - \mathbf{f}_n) \tag{7.107}$$

and

$$\tilde{r}^{ij} = r_n^{ij} + W_1 \Theta_1 \dot{r}_n^{ij} \Delta t + W_2 \Theta_2 \ddot{r}_n^{ij} \Delta t^2 + W_3 \Theta_3 \left(\ddot{r}_{n+1}^{ij} - \ddot{r}_n^{ij} \right) \Delta t^2 \tag{7.108}$$

The corresponding generalized updates are:

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \lambda_1 \dot{\mathbf{q}}_n \Delta t + \lambda_2 \ddot{\mathbf{q}}_n \Delta t^2 + \lambda_3 \Delta \mathbf{a} \Delta t^2 \tag{7.109}$$

$$\dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_n + \lambda_4 \ddot{\mathbf{q}}_n \Delta t + \lambda_5 \Delta \mathbf{a} \Delta t \tag{7.110}$$

with $\Delta \mathbf{a} := \ddot{\mathbf{q}}_{n+1} - \ddot{\mathbf{q}}_n$. Similarly, the inter-particle distance at time t_{n+1} may be approximately obtained from

$$r_{n+1}^{ij} = r_n^{ij} + \theta_1 \dot{r}_n^{ij} \Delta t + \theta_2 \ddot{r}_n^{ij} \Delta t^2 + \theta_3 \left(\ddot{r}_{n+1}^{ij} - \ddot{r}_n^{ij} \right) \Delta t^2 \tag{7.111}$$

Substituting Eq. (7.111) into Eq. (7.108) yields

$$\begin{aligned} \tilde{r}^{ij} =& r_n^{ij} + \frac{W_3 \Theta_3}{\theta_3} (r_{n+1}^{ij} - r_n^{ij}) \\ &+ \left(W_1 \Theta_1 - \frac{W_3 \Theta_3 \theta_1}{\theta_3} \right) \dot{r}_n^{ij} \Delta t + \left(W_2 \Theta_2 - \frac{W_3 \Theta_3 \theta_2}{\theta_3} \right) \ddot{r}_n^{ij} \Delta t^2 \end{aligned} \tag{7.112}$$

In the above procedure of the normalized time weighted residual methodology, the unknowns are the approximations in time at

$$\begin{aligned} \mathbf{f}_n &\approx \mathbf{f}(t_n) \quad & \mathbf{f}_{n+1} &\approx \mathbf{f}(t_{n+1}) \\ r_n^{ij} &\approx r^{ij}(t_n) \quad & r_{n+1}^{ij} &\approx r^{ij}(t_{n+1}) \\ \mathbf{q}_n &\approx \mathbf{q}(t_n) \quad & \mathbf{q}_{n+1} &\approx \mathbf{q}(t_{n+1}) \\ \dot{\mathbf{q}}_n &\approx \dot{\mathbf{q}}(t_n) \quad & \dot{\mathbf{q}}_{n+1} &\approx \dot{\mathbf{q}}(t_{n+1}) \\ \ddot{\mathbf{q}}_n &\approx \ddot{\mathbf{q}}(t_{n-\phi}) \quad & \ddot{\mathbf{q}}_{n+1} &\approx \ddot{\mathbf{q}}(t_{n+1-\phi}) \end{aligned} \tag{7.113}$$

where $\phi := W_1(\Lambda_6 - 1)$. The algorithmic spectral conditions are directly employed from Algorithm 2 and Algorithm 3 as the extensions of the U0- and V0-family I-GSSSS algorithms via the normalized time weighted residual methodology, respectively.

Algorithm 22

Single-field Form I-GSSSS U0- and V0-Family of Algorithms for Nonlinear Dynamical Systems: Option III

Integrator:

$$m_i\tilde{\mathbf{a}}^i - \sum_{j\neq i}^{N} \mathscr{V}'_{ij}(\tilde{r}^{ij})\frac{\tilde{\mathbf{q}}^{ij}}{\tilde{r}^{ij}} = \mathbf{f}_i^{\mathrm{e}}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}) + \tilde{\mathbf{f}}_i$$

where

$$\begin{aligned} \tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1\Lambda_6\Delta\mathbf{a} \\ \tilde{\mathbf{v}} &= \dot{\mathbf{q}}_n + W_1\Lambda_4\ddot{\mathbf{q}}_n\Delta t + W_2\Lambda_5\Delta\mathbf{a}\Delta t \\ \tilde{\mathbf{q}} &= \mathbf{q}_n + W_1\Lambda_1\dot{\mathbf{q}}_n\Delta t + W_2\Lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + W_3\Lambda_3\Delta\mathbf{a}\Delta t^2 \\ \tilde{\mathbf{f}} &= (1 - W_1)\mathbf{f}_n + W_1\mathbf{f}_{n+1} \;\; or \;\; \mathbf{f}(t_{n+W_1}) \end{aligned}$$

and

$$\begin{aligned} \tilde{r}^{ij} =& r_n^{ij} + \frac{W_3\Theta_3}{\theta_3}(r_{n+1}^{ij} - r_n^{ij}) \\ &+ \left(W_1\Theta_1 - \frac{W_3\Theta_3\theta_1}{\theta_3}\right)\dot{r}_n^{ij}\Delta t + \left(W_2\Theta_2 - \frac{W_3\Theta_3\theta_2}{\theta_3}\right)\ddot{r}_n^{ij}\Delta t^2 \end{aligned}$$

Updates:

$$\begin{aligned}\mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + \lambda_3\Delta\mathbf{a}\Delta t^2\\ \dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5\Delta\mathbf{a}\Delta t\\ \ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_n + \Delta\mathbf{a}\end{aligned}$$

Initial conditions:

$$\mathbf{q}(t_0) = \mathbf{q}_0 \text{ and } \dot{\mathbf{q}}(t_0) = \dot{\mathbf{q}}_0$$

Algorithmic parameters:

U0 Based-Option III Algorithms: *Employ the U0-family of algorithmic parameters from Algorithm 2*

V0 Based-Option III Algorithms: *Employ the V0-family of algorithmic parameters from Algorithm 3*

Remark 22 (Algorithm 22)

1. ***Time Level Consistency:*** *Algorithm 22 satisfies the balance equation of motion in the discrete system at the algorithmic time level $t^* = t_{n+W_1}$ as*

$$\begin{aligned}\mathbf{0} &= m_i\ddot{\mathbf{q}}^i(t^*) - \sum_{j\neq i}^{N}\mathscr{V}'_{ij}(r^{ij}(t^*))\frac{\mathbf{q}^{ij}(t^*)}{r^{ij}(t^*)} - \mathbf{f}_i^{\text{ext}}(\mathbf{q}(t^*),\dot{\mathbf{q}}(t^*),t^*)\\ &= m_i\tilde{\mathbf{a}}^i - \sum_{j\neq i}^{N}\mathscr{V}'_{ij}(\tilde{r}^{ij})\frac{\tilde{\mathbf{q}}^{ij}}{\tilde{r}^{ij}} - \mathbf{f}_i^{\text{ext}}(\tilde{\mathbf{q}},\tilde{\mathbf{v}},t_{n+W_1}) + \mathcal{O}(\Delta t^2)\end{aligned} \tag{7.114}$$

   Consequently, the second-order time accuracies in the acceleration, velocity, and configuration are guaranteed.

2. *Assuming $\theta_1 = \Theta_1 = 1$, $\theta_2 = \Theta_2$, and $\theta_3 = \Theta_3$, all members in Algorithm 22 possess the second-order time accuracy, and the algorithmic inter-particle distance can be written as*

$$\tilde{r}^{ij} = r_n^{ij} + W_3(r_{n+1}^{ij} - r_n^{ij}) + (W_3 - W_1)\dot{r}_n^{ij}\Delta t \tag{7.115}$$

which can be reduced to

$$\tilde{r}^{ij} = r_n^{ij} + W_1(r_{n+1}^{ij} - r_n^{ij}) =: r_{n+W_1}^{ij} \tag{7.116}$$

for the U0 based-family of algorithms ($W_1 = W_2 = W_3$). And selecting $\rho_\infty^{\min} = \rho_\infty^{\max} = \rho_\infty^{\mathrm{s}} = 1.0$ in the U0-Based family, Algorithm 22, or $\rho_\infty^{\min} = \rho_\infty^{\max} = 1.0$ and $\rho_\infty^{\mathrm{s}} \in [0.0, 1.0]$ in the V0-Based family, Algorithm 22, we can obtain

$$m_i \tilde{\mathbf{a}}_{n+1/2}^i - \sum_{j \neq i}^{N} \mathscr{V}'_{ij}(r_{n+1/2}^{ij}) \frac{\mathbf{q}_{n+1/2}^{ij}}{r_{n+1/2}^{ij}} = \mathbf{f}_i^{\mathrm{e}}(\mathbf{q}_{n+1/2}, \dot{\mathbf{q}}_{n+1/2}) + \mathbf{f}_i(t_{n+1/2}) \tag{7.117}$$

where

$$\tilde{\mathbf{a}}^i = \frac{\rho_\infty^{\mathrm{s}} \ddot{\mathbf{q}}_n^i + \ddot{\mathbf{q}}_{n+1}^i}{1 + \rho_\infty^{\mathrm{s}}} \tag{7.118}$$

with

$$\frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t} = \dot{\mathbf{q}}_n + \frac{\Delta t}{2}\left(\frac{\rho_\infty^{\mathrm{s}} \ddot{\mathbf{q}}_n + \ddot{\mathbf{q}}_{n+1}}{1 + \rho_\infty^{\mathrm{s}}}\right) \tag{7.119}$$

$$\frac{\dot{\mathbf{q}}_{n+1} - \dot{\mathbf{q}}_n}{\Delta t} = \frac{\rho_\infty^{\mathrm{s}} \ddot{\mathbf{q}}_n + \ddot{\mathbf{q}}_{n+1}}{1 + \rho_\infty^{\mathrm{s}}} \tag{7.120}$$

In the conservative system, it reduces to

$$m_i \mathbf{a}_{n+1/2}^i - \sum_{j \neq i}^{N} \mathscr{V}'_{ij}(r_{n+1/2}^{ij}) \frac{\mathbf{q}_n^{ij} + \mathbf{q}_{n+1}^{ij}}{r_n^{ij} + r_{n+1}^{ij}} = \mathbf{0} \tag{7.121}$$

with

$$\frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t} = \dot{\mathbf{q}}_n + \frac{\Delta t}{2}\left(\frac{\rho_\infty^{\mathrm{s}} \ddot{\mathbf{q}}_n + \ddot{\mathbf{q}}_{n+1}}{1 + \rho_\infty^{\mathrm{s}}}\right) \tag{7.122}$$

$$\frac{\dot{\mathbf{q}}_{n+1} - \dot{\mathbf{q}}_n}{\Delta t} = \frac{\rho_\infty^{\mathrm{s}} \ddot{\mathbf{q}}_n + \ddot{\mathbf{q}}_{n+1}}{1 + \rho_\infty^{\mathrm{s}}} \tag{7.123}$$

Note that only when we select $\rho_\infty^{\mathrm{s}} = 0.0$ in Eq. (7.117) - Eq. (7.123), i.e., $\rho_\infty^{\min} = \rho_\infty^{\max} = 1.0$ and $\rho_\infty^{\mathrm{s}} = 0.0$ in the V0-based family, Algorithm 22, which can be regarded as the extension of the midpoint rule with the midpoint acceleration (MPR-MPA), we can obtain that scheme which does

not require the information of the acceleration at previous time step. That is,

$$m_i \ddot{\mathbf{q}}^i_{n+1} - \sum_{j \neq i}^{N} \mathscr{V}'_{ij}(r^{ij}_{n+1/2}) \frac{\mathbf{q}^{ij}_{n+1/2}}{r^{ij}_{n+1/2}} = \mathbf{f}^{\mathrm{e}}_i(\mathbf{q}_{n+1/2}, \dot{\mathbf{q}}_{n+1/2}) + \mathbf{f}_i(t_{n+1/2}) \quad (7.124)$$

with

$$\frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t} = \dot{\mathbf{q}}_n + \frac{\Delta t}{2} \ddot{\mathbf{q}}_{n+1} \quad (7.125)$$

$$\frac{\dot{\mathbf{q}}_{n+1} - \dot{\mathbf{q}}_n}{\Delta t} = \ddot{\mathbf{q}}_{n+1} \quad (7.126)$$

In this case, we have $\ddot{\mathbf{q}}_{n+1} \approx \ddot{\mathbf{q}}(t_{n+1/2})$ *because of* $\phi = 1/2$. *Strictly speaking, the algorithm described by Eq.* (7.124) - *Eq.* (7.126) *with the initial conditions can be considered as the equivalent form of the* ***assumed distance method*** *[53], which was derived in the sense of the two-field form, in the conservative systems. That is, the algorithm described by Eq.* (7.124) - *Eq.* (7.126) *can be considered as the assumed distance method in the sense of the single-field form, and it is the exact energy-momentum conserving scheme for the potential energy* $\mathscr{V}$ *which is given as a polynomial function of degree two or less; see [54]. Note that the assumed distance method conserved the momentum exactly in* $[t_n, t_{n+1}]$ *for general* $\mathscr{V}$.

Exact Energy-Momentum Framework (Modified Option III): The assumed distance method (in the single-field form) are strictly not the ***exact*** energy-momentum conserving schemes. Based on the time level consistency theorem, we extend Algorithm 22 by proposing the replacement

$$\mathscr{V}'_{ij}(\tilde{r}^{ij}) \longrightarrow \widetilde{\mathscr{V}}'_{ij} = \frac{\mathscr{V}_{ij}(r^{ij}_{n+1}) - \mathscr{V}_{ij}(r^{ij}_n)}{r^{ij}_{n+1} - r^{ij}_n} \quad (7.127)$$

when the algorithmic time level becomes $t^* = t_{n+1/2}$, i.e., when the spectral conditions are selected as $\rho^{\min}_\infty = \rho^{\max}_\infty = \rho^{\mathrm{s}}_\infty = 1.0$ in the U0-based family, Algorithm 22 or $\rho^{\min}_\infty = \rho^{\max}_\infty = 1.0$ and $\rho^{\mathrm{s}}_\infty \in [0.0, 1.0]$ in the V0-based family, Algorithm

22. Note that the replacement is valid only when the algorithmic time level is $t^* = t_{n+1/2}$. Notice that

$$\mathscr{V}'_{ij}(r^{ij}(t_{n+W_1})) - \mathscr{V}'_{ij}(\tilde{r}^{ij}) = \mathcal{O}(\Delta t^2) \text{ and } \mathscr{V}'_{ij}(r^{ij}(t_{n+1/2})) - \widetilde{\mathscr{V}}'_{ij} = \mathcal{O}(\Delta t^2) \quad (7.128)$$

Therefore,

$$\mathscr{V}'_{ij}(\tilde{r}^{ij}) - \widetilde{\mathscr{V}}'_{ij} = \mathcal{O}(\Delta t^2) \quad (7.129)$$

only if $W_1 = 1/2$.

Algorithm 23

Single-field Form I-GSSSS U0- and V0-Family of Algorithms for Nonlinear Dynamical Systems: Exact Energy-Momentum Conserving and Dissipative Algorithms

Integrator:

$$m_i\tilde{\mathbf{a}}^i - \sum_{j\neq i}^{N} \widetilde{\mathscr{V}}'_{ij}\frac{\tilde{\mathbf{q}}^{ij}}{\tilde{r}^{ij}} = \mathbf{f}^{\mathrm{e}}_i(\tilde{\mathbf{q}},\tilde{\mathbf{v}}) + \tilde{\mathbf{f}}_i$$

$$\widetilde{\mathscr{V}}'_{ij} = \frac{\mathscr{V}_{ij}(r^{ij}_{n+1}) - \mathscr{V}_{ij}(r^{ij}_n)}{r^{ij}_{n+1} - r^{ij}_n} \;\; \textit{if } t^* = t_{n+1/2};\; \textit{otherwise, } \widetilde{\mathscr{V}}'_{ij} = \mathscr{V}'_{ij}(\tilde{r}^{ij})$$

where

$$\begin{aligned}
\tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1\Lambda_6\Delta\mathbf{a} \\
\tilde{\mathbf{v}} &= \dot{\mathbf{q}}_n + W_1\Lambda_4\ddot{\mathbf{q}}_n\Delta t + W_2\Lambda_5\Delta\mathbf{a}\Delta t \\
\tilde{\mathbf{q}} &= \mathbf{q}_n + W_1\Lambda_1\dot{\mathbf{q}}_n\Delta t + W_2\Lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + W_3\Lambda_3\Delta\mathbf{a}\Delta t^2 \\
\tilde{\mathbf{f}} &= (1-W_1)\mathbf{f}_n + W_1\mathbf{f}_{n+1} \;\; \textit{or} \;\; \mathbf{f}(t_{n+W_1})
\end{aligned}$$

and

$$\begin{aligned}
\tilde{r}^{ij} =& r^{ij}_n + \frac{W_3\Theta_3}{\theta_3}(r^{ij}_{n+1} - r^{ij}_n) \\
&+ \left(W_1\Theta_1 - \frac{W_3\Theta_3\theta_1}{\theta_3}\right)\dot{r}^{ij}_n\Delta t + \left(W_2\Theta_2 - \frac{W_3\Theta_3\theta_2}{\theta_3}\right)\ddot{r}^{ij}_n\Delta t^2
\end{aligned}$$

Updates:

$$\begin{aligned}
\mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_1 \dot{\mathbf{q}}_n \Delta t + \lambda_2 \ddot{\mathbf{q}}_n \Delta t^2 + \lambda_3 \Delta \mathbf{a} \Delta t^2 \\
\dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4 \ddot{\mathbf{q}}_n \Delta t + \lambda_5 \Delta \mathbf{a} \Delta t \\
\ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_n + \Delta \mathbf{a}
\end{aligned}$$

Initial conditions:

$$\mathbf{q}(t_0) = \mathbf{q}_0 \text{ and } \dot{\mathbf{q}}(t_0) = \dot{\mathbf{q}}_0$$

Algorithmic parameters:

U0 Based-Option III Algorithms: *Employ the U0-family of algorithmic parameters from Algorithm 2*

V0 Based-Option III Algorithms: *Employ the V0-family of algorithmic parameters from Algorithm 3*

Remark 23 (Algorithm 23)

1. *Assuming $\theta_1 = \Theta_1 = 1$, $\theta_2 = \Theta_2$, and $\theta_3 = \Theta_3$, all members in Algorithm 23 possess the second-order time accuracy. When the algorithmic time level is $t^* = t_{n+1/2}$, i.e., $\rho_\infty^{\min} = \rho_\infty^{\max} = \rho_\infty^{\mathrm{s}} = 1.0$ in the U0-based family, Algorithm 23 or $\rho_\infty^{\min} = \rho_\infty^{\max} = 1.0$ and $\rho_\infty^{\mathrm{s}} \in [0.0, 1.0]$ in the V0-based family, Algorithm 23, we obtain the* ***exact energy-momentum conserving schemes*** *in the sense of the single-field form (the unconditional stability of the algorithm is guaranteed from the exact energy conserving features):*

$$m_i \tilde{\mathbf{a}}^i - \sum_{j \neq i}^{N} \frac{\mathscr{V}_{ij}(r_{n+1}^{ij}) - \mathscr{V}_{ij}(r_n^{ij})}{r_{n+1}^{ij} - r_n^{ij}} \frac{\mathbf{q}_n^{ij} + \mathbf{q}_{n+1}^{ij}}{r_n^{ij} + r_{n+1}^{ij}} = \mathbf{f}_i^{\mathrm{e}}(\mathbf{q}_{n+1/2}, \dot{\mathbf{q}}_{n+1/2}) + \mathbf{f}_i(t_{n+1/2}) \tag{7.130}$$

where

$$\tilde{\mathbf{a}}^i = \frac{\rho_\infty^{\mathrm{s}} \ddot{\mathbf{q}}_n^i + \ddot{\mathbf{q}}_{n+1}^i}{1 + \rho_\infty^{\mathrm{s}}} \tag{7.131}$$

with

$$\frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t} = \dot{\mathbf{q}}_n + \frac{\Delta t}{2} \left(\frac{\rho_\infty^{\mathrm{s}} \ddot{\mathbf{q}}_n + \ddot{\mathbf{q}}_{n+1}}{1 + \rho_\infty^{\mathrm{s}}} \right) \tag{7.132}$$

$$\frac{\dot{\mathbf{q}}_{n+1} - \dot{\mathbf{q}}_n}{\Delta t} = \frac{\rho_\infty^{\mathrm{s}} \ddot{\mathbf{q}}_n + \ddot{\mathbf{q}}_{n+1}}{1 + \rho_\infty^{\mathrm{s}}} \tag{7.133}$$

and if $t^* \neq t_{n+1/2}$, *we recover Algorithm 22:*

$$m_i\tilde{\mathbf{a}}^i - \sum_{j\neq i}^{N} \mathscr{V}'_{ij}(\tilde{r}^{ij})\frac{\tilde{\mathbf{q}}^{ij}}{\tilde{r}^{ij}} = \mathbf{f}_i^{\mathrm{e}}(\tilde{\mathbf{q}},\tilde{\mathbf{v}}) + \tilde{\mathbf{f}}_i \tag{7.134}$$

with

$$\frac{\mathbf{q}_{n+1}-\mathbf{q}_n}{\Delta t} = \dot{\mathbf{q}}_n + \frac{\Delta t}{2}\left(\ddot{\mathbf{q}}_n + 2\lambda_3\Delta\mathbf{a}\right) \tag{7.135}$$

$$\frac{\dot{\mathbf{q}}_{n+1}-\dot{\mathbf{q}}_n}{\Delta t} = \ddot{\mathbf{q}}_n + \lambda_5\Delta\mathbf{a} \tag{7.136}$$

As discussed in Remark 22, strictly speaking, only when we select $\rho_\infty^{\mathrm{s}} = 0.0$ *in Eq. (7.130) - Eq. (7.133), i.e.,* $\rho_\infty^{\min} = \rho_\infty^{\max} = 1.0$ *and* $\rho_\infty^{\mathrm{s}} = 0.0$ *in the V0-based family, Algorithm 23, which can be regarded as the extension of the midpoint rule with the midpoint acceleration (MPR-MPA), we obtain*

$$m_i\ddot{\mathbf{q}}^i_{n+1} - \sum_{j\neq i}^{N} \frac{\mathscr{V}_{ij}(r^{ij}_{n+1}) - \mathscr{V}_{ij}(r^{ij}_n)}{r^{ij}_{n+1} - r^{ij}_n}\frac{\mathbf{q}^{ij}_n + \mathbf{q}^{ij}_{n+1}}{r^{ij}_n + r^{ij}_{n+1}} = \mathbf{f}_i^{\mathrm{e}}(\mathbf{q}_{n+1/2},\dot{\mathbf{q}}_{n+1/2}) + \mathbf{f}_i(t_{n+1/2}) \tag{7.137}$$

with

$$\frac{\mathbf{q}_{n+1}-\mathbf{q}_n}{\Delta t} = \dot{\mathbf{q}}_n + \frac{\Delta t}{2}\ddot{\mathbf{q}}_{n+1} \tag{7.138}$$

$$\frac{\dot{\mathbf{q}}_{n+1}-\dot{\mathbf{q}}_n}{\Delta t} = \ddot{\mathbf{q}}_{n+1} \approx \ddot{\mathbf{q}}(t_{n+1/2}) \tag{7.139}$$

In the conservative system, it is the equivalent form of the ***exact energy-momentum conserving algorithm in the two-field form*** *from the numerical point of view. However, the mechanical energy in* $[t_n, t_{n+1}]$ *is exactly conserved for* $\rho_\infty^{\mathrm{s}} \in [0.0, 1.0]$ *with* $\rho_\infty^{\min} = \rho_\infty^{\max} = 1.0$ *in the V0-based family, Algorithm 23, for the non-constrained, conservative system.*

7.2 Elastodynamical Systems

Similar to the exact energy-momentum conserving algorithm designs from the principle of balance of discrete mechanical energy in the conservative system for

the N-body system, we can also derive the exact energy-momentum conserving algorithms for the general case of a hyperelastic material model. Then, we can also construct the new framework of the exact energy-momentum conserving implicit GSSSS algorithms via the normalized time weighted residual methodology. Just as we discretized the configuration $\mathbf{q}$ and the inter-particle distance r separately in time in the previous section, here we discretize in a hybrid fashion the configuration and the discrete strain tensor field separately in time.

7.2.1 Two-field Form: Discrete Total Energy Framework in a Conservative System - Energy and Energy-Momentum Conserving Algorithms

Solving Eq. (7.10) with the second-order time accurate approximation $\dot{\mathbf{q}}_{n+1/2} = (\mathbf{q}_{n+1} - \mathbf{q}_n)/\Delta t$, we arrive at the following system of equations:

$$\mathbf{M}\frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t} + \nabla \mathscr{U}^h_{\text{total}}(\mathbf{q}_{n+\beta_0}) = \mathbf{0} \tag{7.140a}$$

$$\frac{\boldsymbol{\nu}_{n+1} + \boldsymbol{\nu}_n}{2} = \frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t} \tag{7.140b}$$

and there exists some $\beta_0 \in (0,1) \subset \mathbb{R}$ such that

$$\mathscr{U}^h_{\text{total}}(\mathbf{q}_{n+1}) - \mathscr{U}^h_{\text{total}}(\mathbf{q}_n) = [\mathbf{q}_{n+1} - \mathbf{q}_n] \cdot \nabla \mathscr{U}^h_{\text{total}}(\mathbf{q}_{n+\beta_0}) \tag{7.140c}$$

Equation (7.140c) shows that we need to evaluate the forces which are derivable from the total potential energy by means of the mean value theorem with respect to the position vectors of the nodal points. The discrete total potential energy is assumed to consist of the discrete internal potential energy and the external potential energy; however, since the prescribed body force vectors $\mathbf{B} : \mathcal{B} \times \mathbb{I} \to \mathbb{R}^{\text{n}_{\text{dim}}}$ and the prescribed surface nominal traction vectors $\bar{\mathbf{T}}$ on $\partial\mathcal{B}_\sigma \times \mathbb{I}$ do not

depend on $\mathbf{q}$, assume $\nabla\mathscr{U}^h_{\text{total}}(\mathbf{q}_{n+\beta_0}) = \nabla\mathscr{U}^h_{\text{int}}(\mathbf{q}_{n+\beta_0})$; therefore,

$$\begin{aligned}\nabla\mathscr{U}^h_{\text{total}}(\mathbf{q}_{n+\beta_0}) &= \nabla\mathscr{U}^h_{\text{int}}(\mathbf{q}_{n+\beta_0}) = \mathbf{F}^{\text{int}^h}_A(\mathbf{q}_{n+\beta_0}) \\ &= \int_{\mathcal{B}} \mathbf{F}^h(\mathbf{q}_{n+\beta_0}) \cdot \hat{\mathbf{S}}^h(\mathbf{C}^h(\mathbf{q}_{n+\beta_0}))\text{GRAD } N_A dV \quad (7.141)\end{aligned}$$

For the discrete applied external force terms, we introduce $\alpha \in \mathbb{R}$ such that

$$\begin{aligned}\nabla\mathscr{U}^h_{\text{ext}}(\mathbf{q}_{n+\beta_0}) &= \frac{\partial}{\partial \mathbf{q}^A}\left(-\sum_{A=1}^{n_{\text{node}}}\left[\int_{\mathcal{B}} N_A\rho_0\mathbf{B}_{n+\alpha}dV + \int_{\partial\mathcal{B}_\sigma} N_A\bar{\mathbf{T}}_{n+\alpha}dA\right]\cdot\mathbf{q}^A\right) \\ &= \mathbf{F}^{\text{ext}^h}_{A_{n+\alpha}} = -\int_{\mathcal{B}} N_A\rho_0\mathbf{B}_{n+\alpha}dV - \int_{\partial\mathcal{B}_\sigma} N_A\bar{\mathbf{T}}_{n+\alpha}dA \quad (7.142)\end{aligned}$$

However, in order to guarantee the second-order time accuracy of the algorithm, we must have $\alpha = \beta_0 = 1/2$.

Algorithm 24

Two-Field Form Energy Conserving Algorithm with $\beta-$iteration

Maps $TQ \to TQ$ given by $(\mathbf{q}_n, \boldsymbol{\nu}_n) \mapsto (\mathbf{q}_{n+1}, \boldsymbol{\nu}_{n+1})$ which satisfies

$$\sum_{B=1}^{n_{\text{node}}} M_{AB}\frac{\boldsymbol{\nu}^B_{n+1} - \boldsymbol{\nu}^B_n}{\Delta t} + \int_{\mathcal{B}} \mathbf{F}^h(\mathbf{q}_{n+\beta_0}) \cdot \hat{\mathbf{S}}^h(\mathbf{C}^h(\mathbf{q}_{n+\beta_0}))\text{GRAD } N_A dV = \mathbf{0}$$

$$\frac{\boldsymbol{\nu}_{n+1} + \boldsymbol{\nu}_n}{2} = \frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t}$$

and there exists some $\beta_0 \in (0,1) \subset \mathbb{R}$ such that

$$\mathscr{U}^h_{\text{total}}(\mathbf{q}_{n+1}) - \mathscr{U}^h_{\text{total}}(\mathbf{q}_n) = [\mathbf{q}_{n+1} - \mathbf{q}_n] \cdot \nabla\mathscr{U}^h_{\text{total}}(\mathbf{q}_{n+\beta_0})$$

Remark 24 (Algorithm 24)

1. *The discrete right Cauchy-Green deformation tensor $\mathbf{C}^h(\mathbf{q}_{n+\beta_0})$ is symmetric and positive-definite for any $\beta_0 \in (0,1)$:*

$$\begin{aligned}\mathbf{C}^{h^T}(\mathbf{q}_{n+\beta_0}) &= \left(\mathbf{F}^{h^T}(\mathbf{q}_{n+\beta_0}) \cdot \mathbf{F}^h(\mathbf{q}_{n+\beta_0})\right)^T \\ &= \mathbf{F}^{h^T}(\mathbf{q}_{n+\beta_0}) \cdot \mathbf{F}^h(\mathbf{q}_{n+\beta_0}) = \mathbf{C}^h(\mathbf{q}_{n+\beta_0}) \quad (7.144)\end{aligned}$$

and

$$\begin{aligned}\boldsymbol{w} \cdot \mathbf{C}^h(\mathbf{q}_{n+\beta_0})\boldsymbol{w} &= \boldsymbol{w} \cdot \mathbf{F}^{h^T}(\mathbf{q}_{n+\beta_0})\mathbf{F}^h(\mathbf{q}_{n+\beta_0})\boldsymbol{w} \\ &= \mathbf{F}^h(\mathbf{q}_{n+\beta_0})\boldsymbol{w} \cdot \mathbf{F}^h(\mathbf{q}_{n+\beta_0})\boldsymbol{w} \\ &= \| \mathbf{F}^h(\mathbf{q}_{n+\beta_0})\boldsymbol{w} \|^2 > 0 \quad (7.145)\end{aligned}$$

where $\boldsymbol{w} \in \mathbb{R}^{\mathrm{n_{dof}}}$ *is an arbitrary vector, respectively. The corresponding discrete Lagrange-Green strain tensor is calculated by*

$$\mathbf{E}^h(\mathbf{q}_{n+\beta_0}) := \tfrac{1}{2}\left[\mathbf{C}^h(\mathbf{q}_{n+\beta_0}) - \mathbf{I}\right] \quad \textit{with} \quad \beta_0 \in (0,1) \subset \mathbb{R} \tag{7.146}$$

and the algorithmic second Piola-Kirchhoff stress tensor

$$\tilde{\mathbf{S}}^h := \mathbf{S}^h(\mathbf{C}^h(\mathbf{q}_{n+\beta_0})) = \frac{\partial \hat{W}(\mathbf{E}^h(\mathbf{q}_{n+\beta_0}))}{\partial \mathbf{E}^h(\mathbf{q}_{n+\beta_0})} = 2\frac{\partial \hat{W}(\mathbf{C}^h(\mathbf{q}_{n+\beta_0}))}{\partial \mathbf{C}^h(\mathbf{q}_{n+\beta_0})} \tag{7.147}$$

remains symmetric, i.e., $\tilde{\mathbf{S}}^{h^T} = \tilde{\mathbf{S}}^h$.

2. ***Symplecticness****: When* $\beta_0 = 1/2$*, Algorithm 24 becomes the symplectic midpoint rule in the absence of the discrete total external forces,; and it is the only symplectic scheme in the framework of Algorithm 24. In order to study the property of symplecticness of Algorithm 24, consider the system of equations of Algorithm 24 as a mapping* $\mathfrak{G} : T^*Q \to T^*Q$ *on the nodal cotangent bundle*

$$\mathfrak{G}(\mathbf{z}_{n+1}, \mathbf{z}_n) := \mathbf{z}_{n+1} - \mathbf{z}_n - \Delta t \mathbb{J} D\mathscr{H}^h(\mathbf{z}_{n+\beta_0}) \tag{7.148}$$

where the $(2n_{\mathrm{dof}} \times 2n_{\mathrm{dof}})$ *skew-symmetric symplectic matrix*[1] $\mathbb{J}$ *and vector* $\mathbf{z} : \mathbb{I} \to \mathbb{R}^{2\mathrm{n_{dof}}}$ *are defined as*

$$\mathbb{J} := \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{I} & \mathbf{0} \end{bmatrix} \quad \textit{and} \quad \mathbf{z} := \begin{Bmatrix} \mathbf{q} \\ \mathbf{p} \end{Bmatrix} = \begin{Bmatrix} \mathbf{q} \\ \mathbf{M}\boldsymbol{\nu} \end{Bmatrix} \tag{7.150}$$

respectively. Also note

$$D\mathscr{H}^h(\mathbf{z}_{n+\beta_0}) := \begin{Bmatrix} \partial_{\mathbf{q}_{n+\beta_0}} \mathscr{H}^h \\ \partial_{\mathbf{p}_{n+\beta_0}} \mathscr{H}^h \end{Bmatrix} = \begin{Bmatrix} \partial_{\mathbf{q}_{n+\beta_0}} \mathscr{U}^h_{\mathrm{int}} \\ \partial_{\mathbf{p}_{n+\beta_0}} \mathscr{T}^h \end{Bmatrix} = \begin{Bmatrix} \nabla \mathscr{U}^h_{\mathrm{int}}(\mathbf{q}_{n+\beta_0}) \\ \mathbf{M}^{-1}\mathbf{p}_{n+\beta_0} \end{Bmatrix} \tag{7.151}$$

[1] *The symplectic matrix* $\mathbb{J}$ *satisfies*

$$\mathbb{J} = -\mathbb{J}^T = -\mathbb{J}^{-1} \quad \textit{and} \quad \det[\mathbb{J}] = 1 \tag{7.149}$$

Here, $\mathbf{p} = (\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_{n_{\text{node}}}) \in T_q^* Q \equiv \mathbb{R}^{\text{n}_{\text{dof}}}$ *is the set of the nodal (generalized) momentum with* $\mathbf{p}_A = \sum_{B=1}^{n_{\text{node}}} M_{AB} \boldsymbol{\nu}^B \in \mathbb{R}^{\text{n}_{\text{dim}}}$ *in our assumptions; therefore,* $\mathscr{H}^h(\mathbf{q}_{n+\beta_0}, \mathbf{p}_{n+\beta_0}) \in T^*Q \equiv \mathbb{R}^{2\text{n}_{\text{dof}}}$ *denotes the discrete Hamiltonian which is identical to the discrete total energy defined on the nodal cotangent bundle. So that the mapping* $\mathfrak{G}$ *is a symplectic transformation, i.e., for Algorithm 24 to be a symplectic scheme, the following condition must be satisfied; see [55] for details:*

$$\mathbf{A}^T \mathbb{J} \mathbf{A} = \mathbb{J} \Leftrightarrow \mathbf{A}_1 \mathbb{J} \mathbf{A}_1^T = \mathbf{A}_2 \mathbb{J} \mathbf{A}_2^T \tag{7.152}$$

where the linearized amplification matrix $\mathbf{A}$ *is defined by*[2]

$$\delta \mathbf{z}_{n+1} = \mathbf{A} \delta \mathbf{z}_n = \left(\mathbf{A}_1^{-1} \mathbf{A}_2 \right) \delta \mathbf{z}_n \tag{7.154}$$

with

$$\mathbf{A}_1 := \frac{\partial \mathfrak{G}}{\partial \mathbf{z}_{n+1}} = \mathbf{I} - \beta_0 \Delta t \mathbb{J} D^2 \mathscr{H}^h(\mathbf{z}_{n+\beta_0}) \tag{7.155}$$

$$\mathbf{A}_2 := -\frac{\partial \mathfrak{G}}{\partial \mathbf{z}_n} = \mathbf{I} + (1 - \beta_0) \Delta t \mathbb{J} D^2 \mathscr{H}^h(\mathbf{z}_{n+\beta_0}) \tag{7.156}$$

By substitution,

$$\mathbf{A}_1 \mathbb{J} \mathbf{A}_1^T - \mathbf{A}_2 \mathbb{J} \mathbf{A}_2^T = (2\beta_0 - 1) \Delta t^2 \mathbb{J} D^2 \mathscr{H}^h(\mathbf{z}_{n+\beta_0}) \mathbb{J} D^2 \mathscr{H}^h(\mathbf{z}_{n+\beta_0}) \mathbb{J}^T \tag{7.157}$$

Hence, we can clearly see that the midpoint rule ($\beta_0 = 1/2$) *is the only symplectic scheme in the framework of Algorithm 24. Note the properties of symplecticness and energy conservation in* $[t_n, t_{n+1}]$ *in Algorithm 24 cannot co-exist unless the system is linear (where the internal potential energy is given quadratic in* $\mathbf{q}$*) with no external force applied.*

---

[2] *The relation* $\delta \mathbf{z}_{n+1} = \mathbf{A} \delta \mathbf{z}_n$ *for a symplectic transformation implies*

$$d\mathbf{p}_{n+1} \wedge d\mathbf{q}_{n+1} = d\mathbf{p}_n \wedge d\mathbf{q}_n \tag{7.153}$$

For further discussions about the symplectic transformations and the symplectic time integrators; see J.M. Sanz-Serna & M.P. Calvo *[56],* E. Hairer *et al. [57], etc.*

3. *The discrete right Cauchy-Green deformation tensor evaluated with* $\mathbf{q}_{n+\beta_0} \in Q$*, i.e.,* $\mathbf{C}^h(\mathbf{q}_{n+\beta_0})$*, is not physical; see below.*

As stated above, evaluating the discrete right Cauchy-Green deformation tensor by $\mathbf{q}_{n+\beta_0} \in Q$ causes an aritificial deformation tensor $\bar{\mathbf{C}}^h$:

$$\begin{aligned}
\mathbf{C}^h(\mathbf{q}_{n+\beta_0}) &:= \mathbf{F}^{h^T}(\mathbf{q}_{n+\beta_0}) \cdot \mathbf{F}^h(\mathbf{q}_{n+\beta_0}) = \mathbf{F}^{h^T}_{n+\beta_0} \cdot \mathbf{F}^h_{n+\beta_0} \\
&= \left[(1-\beta_0)\mathbf{F}^{h^T}_n + \beta_0 \mathbf{F}^{h^T}_{n+1}\right] \cdot \left[(1-\beta_0)\mathbf{F}^h_n + \beta_0 \mathbf{F}^h_{n+1}\right] \\
&= (1-\beta_0)^2 \mathbf{F}^{h^T}_n \cdot \mathbf{F}^h_n + \beta_0^2 \mathbf{F}^{h^T}_{n+1} \cdot \mathbf{F}^h_{n+1} + \\
&\quad \beta_0(1-\beta_0)(\mathbf{F}^{h^T}_n \cdot \mathbf{F}^h_{n+1} + \mathbf{F}^{h^T}_{n+1} \cdot \mathbf{F}^h_n) \\
&= (1-\beta_0)^2 \mathbf{C}^h_n + \beta_0^2 \mathbf{C}^h_{n+1} + \beta_0(1-\beta_0)\mathbf{C}^h_{\mathrm{d}} \qquad (7.158)
\end{aligned}$$

where $\mathbf{C}^h_{\mathrm{d}} := \mathbf{F}^{h^T}_n \cdot \mathbf{F}^h_{n+1} + \mathbf{F}^{h^T}_{n+1} \cdot \mathbf{F}^h_n$. Using $\mathbf{F}^h = \mathbf{R}^h\mathbf{U}^h$ where $\mathbf{R}^h$ and $\mathbf{U}^h$ denote a discrete rotation tensor and a discrete stretch tensor, respectively, we can clearly see $\mathbf{C}^h_{\mathrm{d}}$ contains nonphysical couplings of rotations and stretches; that is,

$$\mathbf{C}^h_{\mathrm{d}} := \mathbf{U}^{h^T}_n \mathbf{R}^{h^T}_n \cdot \mathbf{R}^h_{n+1}\mathbf{U}^h_{n+1} + \mathbf{U}^{h^T}_{n+1}\mathbf{R}^{h^T}_{n+1}\mathbf{R}^h_n\mathbf{U}^h_n \qquad (7.159)$$

where $\mathbf{R}^{h^T}_n \cdot \mathbf{R}^h_{n+1} \neq \mathbf{I}$ and $\mathbf{R}^{h^T}_{n+1} \cdot \mathbf{R}^h_n \neq \mathbf{I}$ in general. To avoid this issue, we consider the direct interpolation of the discrete right Cauchy-Green deformation tensor:

$$\mathbf{C}^h_{n+\xi} := (1-\xi)\mathbf{C}^h_n + \xi \mathbf{C}^h_{n+1} \quad \text{with} \quad \xi \in [0,1] \subset \mathbb{R} \qquad (7.160)$$

Note $\mathbf{C}^h_{n+\xi} \neq \mathbf{C}^h(\mathbf{q}_{n+\xi})$ for $\xi \in (0,1)$. Using $\mathbf{R}^h$ and $\mathbf{U}^h$, $\mathbf{C}^h_{n+\xi}$ can be written as

$$\begin{aligned}
\mathbf{C}^h_{n+\xi} &= (1-\xi)\mathbf{F}^{h^T}_n \cdot \mathbf{F}^h_n + \xi \mathbf{F}^{h^T}_{n+1} \cdot \mathbf{F}^h_{n+1} \\
&= (1-\xi)\mathbf{U}^{h^T}_n \mathbf{R}^{h^T}_n \cdot \mathbf{R}^h_n \mathbf{U}^h_n + \xi \mathbf{U}^{h^T}_{n+1}\mathbf{R}^{h^T}_{n+1} \cdot \mathbf{R}^h_{n+1}\mathbf{U}^h_{n+1} \\
&= (1-\xi)\mathbf{U}^{h^T}_n \cdot \mathbf{U}^h_n + \xi \mathbf{U}^{h^T}_{n+1} \cdot \mathbf{U}^h_{n+1} \qquad (7.161)
\end{aligned}$$

Notice $\mathbf{C}^h_{n+\xi}$ also maintains the properties of symmetry and positive definiteness for any $\xi \in [0,1]$; that is,

$$\begin{aligned}
\mathbf{C}^{h^T}_{n+\xi} &= \left((1-\xi)\mathbf{F}^{h^T}_n \cdot \mathbf{F}^h_n + \xi \mathbf{F}^{h^T}_{n+1} \cdot \mathbf{F}^h_{n+1}\right)^T \\
&= (1-\xi)\mathbf{F}^{h^T}_n \cdot \mathbf{F}^h_n + \xi \mathbf{F}^{h^T}_{n+1} \cdot \mathbf{F}^h_{n+1} = \mathbf{C}^h_{n+\xi} \qquad (7.162)
\end{aligned}$$

and

$$\begin{aligned}
\boldsymbol{w}\cdot\mathbf{C}^h_{n+\xi}\boldsymbol{w} &= (1-\xi)\boldsymbol{w}\cdot\mathbf{C}^h_n\boldsymbol{w}+\xi\boldsymbol{w}\cdot\mathbf{C}^h_{n+1}\boldsymbol{w}\\
&= (1-\xi)\boldsymbol{w}\cdot\mathbf{F}^{h^T}_n\mathbf{F}^h_n\boldsymbol{w}+\xi\boldsymbol{w}\cdot\mathbf{F}^{h^T}_{n+1}\mathbf{F}^h_{n+1}\boldsymbol{w}\\
&= (1-\xi)\mathbf{F}^h_n\boldsymbol{w}\cdot\mathbf{F}^h_n\boldsymbol{w}+\xi\mathbf{F}^h_{n+1}\boldsymbol{w}\cdot\mathbf{F}^h_{n+1}\boldsymbol{w}\\
&= (1-\xi)\parallel\mathbf{F}^h_n\boldsymbol{w}\parallel^2+\xi\parallel\mathbf{F}^h_{n+1}\boldsymbol{w}\parallel^2>0 \qquad (7.163)
\end{aligned}$$

where $\boldsymbol{w}\in\mathbb{R}^{\mathrm{n_{dof}}}$ is an arbitrary vector, respectively. By definition, the discrete Lagrange-Green strain tensor may be calculated as

$$\mathbf{E}^h_{n+\xi}:=\tfrac{1}{2}\left[\mathbf{C}^h_{n+\xi}-\mathbf{I}\right]\quad\text{with}\quad\xi\in[0,1]\subset\mathbb{R} \qquad (7.164)$$

Hence, by means of the directly interpolated discrete right Cauchy-Green deformation tensor, we can now rewrite the algorithm with β_0, ξ_0 and α as follows:

$$\sum_{B=1}^{n_{\mathrm{node}}}M_{AB}\frac{\boldsymbol{\nu}^B_{n+1}-\boldsymbol{\nu}^B_n}{\Delta t}+\int_{\mathcal{B}}\mathbf{F}^h(\mathbf{q}_{n+\beta_0})\cdot\check{\mathbf{S}}^h\mathrm{GRAD}\;N_A dV=\mathbf{F}^{\mathrm{ext}^h}_{A_{n+\alpha}} \qquad (7.165a)$$

$$\frac{\boldsymbol{\nu}_{n+1}+\boldsymbol{\nu}_n}{2}=\frac{\mathbf{q}_{n+1}-\mathbf{q}_n}{\Delta t} \qquad (7.165b)$$

where the symmetric algorithmic second Piola-Kirchhoff stress tensor $\check{\mathbf{S}}^h=\check{\mathbf{S}}^{h^T}$ is defined by

$$\check{\mathbf{S}}^h:=\mathbf{S}^h(\mathbf{C}^h_{n+\xi_0})=\frac{\partial\bar{W}(\mathbf{E}^h_{n+\xi_0})}{\partial\mathbf{E}^h_{n+\xi_0}}=2\frac{\partial\hat{W}(\mathbf{C}^h_{n+\xi_0})}{\partial(\mathbf{C}^h_{n+\xi_0})} \qquad (7.165c)$$

Recall that the parameter(s) $\beta_0\in(0,1)$ is(are) calculated via the mean value theorem on the internal potential energy with respect to $\mathbf{q}$; that is,

$$\mathscr{U}^h_{\mathrm{int}}(\mathbf{q}_{n+1})-\mathscr{U}^h_{\mathrm{int}}(\mathbf{q}_n)=[\mathbf{q}_{n+1}-\mathbf{q}_n]\cdot\nabla\mathscr{U}^h_{\mathrm{int}}(\mathbf{q}_{n+\beta_0})\quad\text{for some}\;\;\beta_0\in(0,1) \qquad (7.166)$$

However, introducing $\mathbf{C}^h_{n+\xi_0}$ instead of $\mathbf{C}^h(\mathbf{q}_{n+\beta_0})$ implies that we break a direct relation between $\mathbf{C}^h$ and $\mathbf{q}$ via the discrete deformation gradient tensor in the

fully discretized system; therefore, we may assume that the internal nodal forces in the algorithm can be written as

$$\check{\mathbf{F}}_A^{\text{int}^h} = \sum_{B=1}^{n_{\text{node}}} \int_{\mathcal{B}} \underbrace{\text{GRAD } N_A \cdot \mathbf{S}^h(\mathbf{C}^h_{n+\xi_0}) \text{GRAD } N_B}_{\check{S}_{AB}} dV \mathbf{q}^B_{n+\beta_0} \tag{7.167}$$

where $\mathbf{S}^h(\mathbf{C}^h_{n+\xi_0})$ shows the algorithmic constitutive relation. Hence, we may modify the mean value theorem, Eq. (7.166), as

$$\Delta \mathscr{U}^h_{\text{int}} = [\mathbf{q}_{n+1} - \mathbf{q}_n] \cdot \check{\mathbf{F}}^{\text{int}^h} \tag{7.168}$$

Notice that the left-hand side of Eq. (7.168) shows the exact difference of the internal potential energy in $[t_n, t_{n+1}]$ since we assume $\mathbf{C}^h_{n+\xi} = \mathbf{C}^h(\mathbf{q}_{n+\beta})$ only for $\xi = \beta = 1$ and $\xi = \beta = 0$ in general. That is,

$$\Delta \mathscr{U}^h_{\text{int}} := \int_{\mathcal{B}} \hat{W}^h(\mathbf{C}^h_{n+1}) dV - \int_{\mathcal{B}} \hat{W}^h(\mathbf{C}^h_n) dV \tag{7.169}$$

Discrete Linear/Angular Momentum Conservation: The discrete total linear momentum is defined by $\mathbf{L}^h := \sum_{A=1}^{n_{\text{node}}} \sum_{B=1}^{n_{\text{node}}} M_{AB} \boldsymbol{\nu}^B$; therefore, the difference of the discrete total linear momentum within time step over time step size $\Delta t := t_{n+1} - t_n > 0$ may be written by follows from Eq. (7.165a):

$$\begin{aligned}
\frac{\mathbf{L}^h_{n+1} - \mathbf{L}^h_n}{\Delta t} &= \sum_{A=1}^{n_{\text{node}}} \sum_{B=1}^{n_{\text{node}}} M_{AB} \frac{\boldsymbol{\nu}^B_{n+1} - \boldsymbol{\nu}^B_n}{\Delta t} \\
&= \sum_{A=1}^{n_{\text{node}}} \mathbf{F}^{\text{ext}^h}_{A_{n+\alpha}} - \sum_{A=1}^{n_{\text{node}}} \int_{\mathcal{B}} \mathbf{F}^h(\mathbf{q}_{n+\beta_0}) \cdot \check{\mathbf{S}}^h \text{GRAD } N_A dV \\
&= \sum_{A=1}^{n_{\text{node}}} \mathbf{F}^{\text{ext}^h}_{A_{n+\alpha}} - \underbrace{\sum_{A=1}^{n_{\text{node}}} \sum_{B=1}^{n_{\text{node}}} \int_{\mathcal{B}} \text{GRAD } N_A \cdot \check{\mathbf{S}}^h \text{GRAD } N_B dV \mathbf{q}^B_{n+\beta_0}}_{\mathbf{0}} \\
&= \sum_{A=1}^{n_{\text{node}}} \mathbf{F}^{\text{ext}^h}_{A_{n+\alpha}}
\end{aligned} \tag{7.170}$$

Remember the completeness condition of the nodal shape function; that is,

$$\sum_{A=1}^{n_{\text{node}}} N_A = 1 \tag{7.171}$$

Therefore, for the Neumann problem, ($\partial\mathcal{B}_\varphi = 0$), in the absence of the external (applied) nodal forces, the discrete total linear momentum within time step $[t_n, t_{n+1}]$ is exactly conserved for any $\beta_0 \in [0,1]$ and $\xi \in [0,1]$; that is, $(\mathbf{L}^h_{n+1} - \mathbf{L}^h_n)/\Delta t = \mathbf{0}$.

Similarly, in view of the definition of the discrete total angular momentum which is defined by $\mathbf{J}^h := \sum_{A=1}^{n_{\text{node}}} \mathbf{q}^A \times (\sum_{B=1}^{n_{\text{node}}} M_{AB}\boldsymbol{\nu}^B)$, the difference of the discrete total angular momentum within time step over time step size may be written as follows:

$$
\begin{aligned}
\frac{\mathbf{J}^h_{n+1} - \mathbf{J}^h_n}{\Delta t} &= \sum_{A=1}^{n_{\text{node}}} \mathbf{q}^A_{n+1/2} \times \sum_{B=1}^{n_{\text{node}}} M_{AB} \frac{\boldsymbol{\nu}^B_{n+1} - \boldsymbol{\nu}^B_n}{\Delta t} \\
&= \underbrace{\sum_{A=1}^{n_{\text{node}}} \mathbf{q}^A_{n+1/2} \times \mathbf{F}^{\text{ext}^h}_{A_{n+1/2}}}_{\mathbf{T}_{n+1/2}} \\
&\quad - \sum_{A=1}^{n_{\text{node}}} \mathbf{q}^A_{n+1/2} \times \int_{\mathcal{B}} \mathbf{F}^h(\mathbf{q}_{n+1/2}) \cdot \check{\mathbf{S}}^h \text{GRAD } N_A dV \\
&= \mathbf{T}_{n+1/2} - \sum_{A=1}^{n_{\text{node}}} \sum_{B=1}^{n_{\text{node}}} \mathbf{q}^A_{n+1/2} \times \int_{\mathcal{B}} \text{GRAD } N_A \cdot \check{\mathbf{S}}^h \text{GRAD } N_B dV \mathbf{q}^B_{n+1/2} \\
&= \mathbf{T}_{n+1/2} - \underbrace{\sum_{A=1}^{n_{\text{node}}} \sum_{B=1}^{n_{\text{node}}} \int_{\mathcal{B}} \overbrace{\text{GRAD } N_A \cdot \check{\mathbf{S}}^h \text{GRAD } N_B}^{\text{symmetry}} dV \overbrace{\mathbf{q}^A_{n+1/2} \times \mathbf{q}^B_{n+1/2}}^{\text{skew-symmetry}}}_{\mathbf{0}} \\
&= \mathbf{T}_{n+1/2}
\end{aligned}
\tag{7.172}
$$

where $\mathbf{T}$ denotes the resultant external torque. Therefore, because the algorithmic second Piola-Kirchhoff stress tensor $\check{\mathbf{S}}^h$ is symmetric, for the Neumann problem, ($\partial\mathcal{B}_\varphi = 0$), in the absence of the external (applied) nodal forces, the discrete total angular momentum within time step $[t_n, t_{n+1}]$ is exactly conserved; that is, $(\mathbf{J}^h_{n+1} - \mathbf{J}^h_n)/\Delta t = \mathbf{0}$.

The linear momentum conservation in $[t_n, t_{n+1}]$ is guaranteed for all $\beta_0 \in [0,1]$ with arbitrary $\check{\mathbf{S}}^h := \mathbf{S}^h(\mathbf{C}^h_{n+\xi})$, i.e., for any $\xi \in [0,1]$, and the angular momentum conservation in $[t_n, t_{n+1}]$ is guaranteed only for $\beta_0 = 1/2$ with an arbitrary and

symmetric $\check{\mathbf{S}}^h := \mathbf{S}^h(\mathbf{C}^h_{n+\xi})$. Remember $\mathbf{S}^h(\mathbf{C}^h_{n+\xi_0})$ is symmetric for all $\xi \in [0,1]$. Hence, ξ_0 needs to be determined to guarantee the total energy conservation in $[t_n, t_{n+1}]$ with fixed $\beta_0 = 1/2$ at every time step.

Discrete Total Energy Conservation: Premultiplying by $(\mathbf{q}_{n+1} - \mathbf{q}_n)$ on Eq. (7.177a) yields

$$\begin{aligned} &\underbrace{\sum_{A=1}^{n_{\text{node}}} \sum_{B=1}^{n_{\text{node}}} M_{AB} \frac{\boldsymbol{\nu}^B_{n+1} - \boldsymbol{\nu}^B_n}{\Delta t} \cdot (\mathbf{q}^A_{n+1} - \mathbf{q}^A_n)}_{\text{Term 1}} \\ &+ \underbrace{\sum_{A=1}^{n_{\text{node}}} \int_{\mathcal{B}} \mathbf{F}^h(\mathbf{q}_{n+1/2}) \cdot \check{\mathbf{S}}^h \text{GRAD } N_A dV \cdot (\mathbf{q}^A_{n+1} - \mathbf{q}^A_n)}_{\text{Term 2}} \\ &= \sum_{A=1}^{n_{\text{node}}} \mathbf{F}^{\text{ext}^h}_{A_{n+1/2}} \cdot (\mathbf{q}^A_{n+1} - \mathbf{q}^A_n) \end{aligned} \tag{7.173}$$

Now, carefully consider Terms 1 and 2 in Eq. (7.173):
Term 1 yields the difference of the discrete kinetic energy within time step $[t_n, t_{n+1}]$ as

$$\begin{aligned} \text{term 1} &= \sum_{A=1}^{n_{\text{node}}} \sum_{B=1}^{n_{\text{node}}} M_{AB} (\boldsymbol{\nu}^B_{n+1} - \boldsymbol{\nu}^B_n) \cdot \frac{\mathbf{q}^A_{n+1} - \mathbf{q}^A_n}{\Delta t} \\ &= \sum_{A=1}^{n_{\text{node}}} \sum_{B=1}^{n_{\text{node}}} M_{AB} (\boldsymbol{\nu}^B_{n+1} - \boldsymbol{\nu}^B_n) \cdot \frac{\boldsymbol{\nu}^A_{n+1} + \boldsymbol{\nu}^A_n}{2} = \Delta \mathscr{T}^h \end{aligned} \tag{7.174}$$

where Eq. (7.177b) has been used.

Next, Term 2 yields

$$\begin{aligned}
\text{term 2} &= \sum_{A=1}^{n_{\text{node}}} \int_{\mathcal{B}} F^h_{iK} \check{S}^h_{IK} (\mathbf{e}_i \otimes \mathbf{E}_I) \text{GRAD } N^L_A \mathbf{E}_L dV \cdot \Delta q^A_j \mathbf{e}_j \\
&= \sum_{A=1}^{n_{\text{node}}} \int_{\mathcal{B}} F^h_{iK} \check{S}^h_{IK} \text{GRAD } N^L_A \Delta q^A_j \delta_{IL} \delta_{ij} dV \\
&= \sum_{A=1}^{n_{\text{node}}} \int_{\mathcal{B}} F^h_{iK} \check{S}^h_{LK} \text{GRAD } N^L_A \Delta q^A_i dV \\
&= \int_{\mathcal{B}} F^h_{iK} \check{S}^h_{LK} F^h_{iL} dV = \int_{\mathcal{B}} \mathbf{F}^h \check{\mathbf{S}}^h : \mathbf{F}^h dV \\
&= \int_{\mathcal{B}} \check{S}^h_{KL} F^h_{iK} F^h_{iL} dV = \int_{\mathcal{B}} \check{\mathbf{S}}^h : (\mathbf{F}^{h^T} \cdot \mathbf{F}^h) dV \\
&= \int_{\mathcal{B}} \check{\mathbf{S}}^h : \mathbf{C}^h dV \qquad (7.175)
\end{aligned}$$

Hence, we can see that Eq. (7.173) yields the discrete energy conservation in the sense of $\mathscr{E}^h_{n+1} = \mathscr{E}^h_n$ if and only if $\check{\mathbf{S}}^h$ in Eq. (7.175) satisfies the difference of the discrete strain energy within time step $[t_n, t_{n+1}]$, i.e., $\Delta \mathscr{U}^h_{\text{int}}$.

Algorithm 25

Two-Field Form Energy-Momentum Conserving Algorithm

Map $TQ \to TQ$ given by $(\mathbf{q}_n, \boldsymbol{\nu}_n) \mapsto (\mathbf{q}_{n+1}, \boldsymbol{\nu}_{n+1})$ which satisfies

$$\begin{aligned}
&\sum_{B=1}^{n_{\text{node}}} M_{AB} \frac{\boldsymbol{\nu}^B_{n+1} - \boldsymbol{\nu}^B_n}{\Delta t} + \int_{\mathcal{B}} \mathbf{F}^h(\mathbf{q}_{n+1/2}) \cdot \check{\mathbf{S}}^h \text{GRAD } N_A dV = \mathbf{F}^{\text{ext}^h}_{A_{n+1/2}} \\
&\frac{\boldsymbol{\nu}_{n+1} + \boldsymbol{\nu}_n}{2} = \frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t}
\end{aligned}$$

Also, the second Piola-Kirchhoff stress tensor in the algorithm, $\check{\mathbf{S}}^h$, defined by

$$\sum_{B=1}^{n_{\text{node}}} M_{AB} \frac{\boldsymbol{\nu}^B_{n+1} - \boldsymbol{\nu}^B_n}{\Delta t} + \int_{\mathcal{B}} \mathbf{F}^h(\mathbf{q}_{n+1/2}) \cdot \check{\mathbf{S}}^h_{\text{m}} \text{GRAD } N_A dV = \mathbf{F}^{\text{ext}^h}_{A_{n+1/2}} \qquad (7.177a)$$

$$\frac{\boldsymbol{\nu}_{n+1} + \boldsymbol{\nu}_n}{2} = \frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t} \qquad (7.177b)$$

and the modified second Piola-Kirchhoff stress tensor $\check{\mathbf{S}}^h_{\mathrm{m}}$ in the algorithm is defined by

$$\check{\mathbf{S}}^h_{\mathrm{m}} := 2D\hat{W}(\mathbf{C}_{n+1/2}) + 2\frac{\hat{W}(\mathbf{C}^h_{n+1}) - \hat{W}(\mathbf{C}^h_n) - D\hat{W}(\mathbf{C}_{n+1/2}) : \Delta\mathbf{C}^h}{\parallel \Delta\mathbf{C}^h \parallel^2}\Delta\mathbf{C}^h \tag{7.177c}$$

where $\Delta\mathbf{C}^h := \mathbf{C}^h_{n+1} - \mathbf{C}^h_n$ and $\parallel \Delta\mathbf{C}^h \parallel^2 := \Delta\mathbf{C}^h : \Delta\mathbf{C}^h$. The algorithmic second Piola-Kirchhoff stress tensor may be also expressed as

$$\begin{aligned}\check{\mathbf{S}}^h_{\mathrm{m}} &= \underbrace{\left[S^h_{IJ}(\mathbf{C}_{n+1/2}) + \frac{\hat{W}(\mathbf{C}_{n+1}) - \hat{W}(\mathbf{C}_n) - S^h_{KL}(\mathbf{C}_{n+1/2})\Delta C^h_{KL}}{\Delta C^h_{MN}\Delta C^h_{MN}}\Delta C^h_{IJ}\right]}_{\check{S}^h_{IJ}}(\mathbf{E}_I \otimes \mathbf{E}_J) \\ &= \check{S}^h_{IJ}(\mathbf{E}_I \otimes \mathbf{E}_J)\end{aligned} \tag{7.178}$$

It is very important to note that the algorithmic second Piola-Kirchhoff stress tensor $\check{\mathbf{S}}^h_{\mathrm{m}}$ remains symmetric, i.e., $\check{\mathbf{S}}^h_{\mathrm{m}} = \check{\mathbf{S}}^{h^T}_{\mathrm{m}}$ or $\check{S}^h_{IJ} = \check{S}^h_{JI}$ because of $\mathbf{S}^h = \mathbf{S}^{h^T}$ and $\mathbf{C}^h = \mathbf{C}^{h^T}$.

Discrete Total Energy Conservation: Substituting Eq. (7.178) into Eq. (7.175), we get

$$\begin{aligned}\text{Term 2} &= \int_{\mathcal{B}} \left[S^h_{IJ} + \frac{\hat{W}(\mathbf{C}_{n+1}) - \hat{W}(\mathbf{C}_n) - S^h_{KL}\Delta C^h_{KL}}{\Delta C^h_{MN}\Delta C^h_{MN}}\Delta C^h_{IJ}\right] \\ &\quad (\mathbf{E}_I \otimes \mathbf{E}_J) : C^h_{PQ}(\mathbf{E}_P \otimes \mathbf{E}_Q)dV \\ &= \int_{\mathcal{B}} \left[S^h_{IJ} + \frac{\hat{W}(\mathbf{C}_{n+1}) - \hat{W}(\mathbf{C}_n) - S^h_{KL}\Delta C^h_{KL}}{\Delta C^h_{MN}\Delta C^h_{MN}}\Delta C^h_{IJ}\right] C^h_{PQ}\delta_{IP}\delta_{JQ}dV \\ &= \int_{\mathcal{B}} \left[S^h_{IJ} + \frac{\hat{W}(\mathbf{C}_{n+1}) - \hat{W}(\mathbf{C}_n) - S^h_{KL}\Delta C^h_{KL}}{\Delta C^h_{MN}\Delta C^h_{MN}}\Delta C^h_{IJ}\right] C^h_{IJ}dV \\ &= \int_{\mathcal{B}} \hat{W}(\mathbf{C}_{n+1})dV - \int_{\mathcal{B}} \hat{W}(\mathbf{C}_n)dV = \Delta\mathscr{U}^h_{\mathrm{int}}\end{aligned} \tag{7.179}$$

where $S^h_{IJ} = S^h_{IJ}(\mathbf{C}_{n+1/2})$ and $C^h_{IJ} = C^h_{n+1/2\,IJ}$. Therefore, Term 2 shows the difference of the discrete internal potential energy within time step $[t_n, t_{n+1}]$ (notice

that Term 1 yields the difference of the discrete kinetic energy). Hence, the system given by Eq. (7.177) implies

$$\mathscr{E}^h_{n+1} - \mathscr{E}^h_n = 0 \tag{7.180}$$

for the Neumann problem, ($\partial\mathcal{B}_\varphi = 0$), in the absence of the external (applied) nodal forces[3] .

7.2.2 Single-field Form: Extensions to Nonlinear Elastodynamics via Normalized Time Weighted Residual Methodology - Energy-Momentum I-GSSSS Framework

Consider the nonlinear dynamical single-field form system,

$$\boxed{\begin{array}{l} \textbf{\textit{Balance Equation:}} \\ \displaystyle\sum_{j=1}^{n_{\text{node}}} M_{ij}\ddot{\mathbf{q}}^j(t) + \mathbf{F}_i^{\text{int}^h} = \mathbf{F}_i^{\text{ext}^h}(\mathbf{q},\dot{\mathbf{q}},t) \\ \textbf{\textit{Initial conditions:}} \\ \mathbf{q}(t_0) = \mathbf{q}_0 \\ \boldsymbol{\nu}(t_0) = \boldsymbol{\nu}_0 \end{array}} \tag{7.181}$$

where the discrete internal force vector is given by Eq. (6.154), and the discrete total external force is given by

$$\mathbf{F}_i^{\text{ext}^h}(\mathbf{q},\dot{\mathbf{q}},t) = \mathbf{F}_i^{\text{diss}^h}(\mathbf{q},\dot{\mathbf{q}}) + \int_{\mathcal{B}} N_i\rho_0\mathbf{B}(t_{n+W_1})dV + \int_{\partial\mathcal{B}_\sigma} N_i\bar{\mathbf{T}}(t_{n+W_1})dA \tag{7.182}$$

where $\mathbf{F}_i^{\text{diss}^h}(\mathbf{q},\dot{\mathbf{q}})$ denotes the discrete dissipative force vector. Apply the ***normalized time weighted residual methodology***, based on the configuration $\mathbf{q}$ and the discrete right Cauchy-Green deformation tensor $\mathbf{C}^h$, the fully-discretized

[3] More correctly, the boundary condition and the external (applied) nodal forces must be time-independent.

balance equation is obtained as

$$\sum_{j=1}^{n_{\text{node}}} M_{ij}\tilde{\mathbf{a}} + \sum_{j=1}^{n_{\text{node}}} \int_{\mathcal{B}} \text{GRAD } N_i \cdot \mathbf{S}^h(\tilde{\mathbf{C}}^h)\text{GRAD } N_j dV \tilde{\mathbf{q}}^j \cong \mathbf{F}_i^{\text{ext}^h}(\tilde{\mathbf{q}}, \tilde{\boldsymbol{\nu}}, t_{n+W_1}) \tag{7.183}$$

where the algorithmic configuration, velocity, and acceleration vectors are defined in Eq. (6.139)-Eq. (6.141), respectively; and we define the algorithmic discrete right Cauchy-Green deformation tensor $\tilde{\mathbf{C}}^h$ as

$$\tilde{\mathbf{C}}^h = \mathbf{C}_n^h + W_3(\mathbf{C}_{n+1}^h - \mathbf{C}_n^h) + (W_1 - W_3)\dot{\mathbf{C}}_n \Delta t \tag{7.184}$$

where $\mathbf{C}_n^h \approx \mathbf{C}^h(t_n)$ and $\mathbf{C}_{n+1}^h \approx \mathbf{C}^h(t_{n+1})$ in view of Eq. (7.115). For both U0 and V0 family-based GSSSS algorithms, we have $W_3 = W_2$. For the U0 family, we have $W1 = W_2 = W_3$; hence, Eq. (7.184) yields

$$\tilde{\mathbf{C}}^h = \mathbf{C}_n^h + W_1(\mathbf{C}_{n+1}^h - \mathbf{C}_n^h) =: \mathbf{C}_{n+W_1}^h \tag{7.185}$$

Assuming $\mathbf{C}_n^h \approx \mathbf{C}^h(t_n)$, $\dot{\mathbf{C}}_n^h \approx \dot{\mathbf{C}}^h(t_n)$, and $\mathbf{C}_{n+1}^h \approx \mathbf{C}^h(t_{n+1})$, Eq. (7.184) yields

$$\begin{aligned} \tilde{\mathbf{C}}^h &\approx \mathbf{C}^h(t_n) + W_3(\mathbf{C}^h(t_{n+1}) - \mathbf{C}^h(t_n)) + (W_1 - W_3)\dot{\mathbf{C}}(t_n)\Delta t \\ &= \mathbf{C}^h(t_n) + W_1\dot{\mathbf{C}}(t_n)\Delta t + \frac{W_3}{2}\ddot{\mathbf{C}}(t_n)\Delta t^2 + \mathcal{O}(\Delta t^3) \end{aligned} \tag{7.186}$$

Hence,

$$\begin{aligned} &\mathbf{C}^h(t_{n+W_1}) - \left[\mathbf{C}^h(t_n) + W_1\dot{\mathbf{C}}(t_n)\Delta t + \frac{W_3}{2}\ddot{\mathbf{C}}(t_n)\Delta t^2\right] + \mathcal{O}(\Delta t^3) \\ &= \left(W_1^2 - \frac{W_3}{2}\right)\ddot{\mathbf{C}}(t_n)\Delta t^2 + \mathcal{O}(\Delta t^3) = \mathcal{O}(\Delta t^2) \end{aligned} \tag{7.187}$$

regardless of the choices of $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\text{s}})$.

Algorithm 26

I-GSSSS Algorithms for Nonlinear Elastodynamics: Option III

Integrator:

$$\sum_{j=1}^{n_{\text{node}}} M_{ij}\tilde{\mathbf{a}} + \int_{\mathcal{B}} \tilde{\mathbf{F}}^h \cdot \hat{\mathbf{S}}^h(\tilde{\mathbf{C}}^h)\ \text{GRAD } N_i\, dV = \mathbf{F}_i^{\text{ext}^h}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}, t_{n+W_1})$$

where

$$\begin{aligned}
\tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1\Lambda_6\Delta\mathbf{a} \\
\tilde{\mathbf{v}} &= \dot{\mathbf{q}}_n + W_1\Lambda_4\ddot{\mathbf{q}}_n\Delta t + W_2\Lambda_5\Delta\mathbf{a}\Delta t \\
\tilde{\mathbf{q}} &= \mathbf{q}_n + W_1\Lambda_1\dot{\mathbf{q}}_n\Delta t + W_2\Lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + W_3\Lambda_3\Delta\mathbf{a}\Delta t^2 \\
\tilde{\mathbf{C}}^h &= \mathbf{C}_n^h + W_1(\mathbf{C}_{n+1}^h - \mathbf{C}_n^h) + (W_1 - W_3)\dot{\mathbf{C}}_n\Delta t
\end{aligned}$$

Updates:

$$\begin{aligned}
\mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + \lambda_3\Delta\mathbf{a}\Delta t^2 \\
\dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5\Delta\mathbf{a}\Delta t \\
\ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_n + \Delta\mathbf{a}
\end{aligned}$$

Initial conditions:

$$\begin{aligned}
\mathbf{q}(t_0) &= \mathbf{q}_0 \\
\dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0
\end{aligned}$$

Algorithmic parameters:

U0 Based-Option III Algorithms: *Employ the U0-family of algorithmic parameters from Algorithm 2*

V0 Based-Option III Algorithms: *Employ the V0-family of algorithmic parameters from Algorithm 3*

Remark 25 (Algorithm 26)

1. *Algorithm 26 is an extension of Algorithms 2 and 3 originally derived for linear elastodynamics to nonlinear elastodynamics. The algorithmic parameters are directly employed from Algorithms 2 and 3 for the U0-family and V0-family based schemes, respectively. The representations yield energy-momentum based algorithm designs for the St. Venant-Kirchhoff material models (and, when controllable numerical dissipation is turned off, they readily yield a family of energy-momentum conserving algorithms in the strict sense of the single-field form of representation of the equation of motion [43, 44, 58]).*

2. *The algorithmic discrete deformation gradient tensor field can be simply given by*

$$\tilde{\mathbf{F}}^h = \mathbf{I} + \sum_{i=1}^{n_{\text{node}}} \tilde{\mathbf{q}}^i \otimes \text{GRAD}\, N_i(\mathbf{X}) \tag{7.188}$$

The discrete internal force vector in the algorithm

$$\begin{aligned} &\int_{\mathcal{B}} \tilde{\mathbf{F}}^h \cdot \hat{\mathbf{S}}^h(\tilde{\mathbf{C}}^h)\ \text{GRAD}\ N_i\, dV \\ &= \sum_{j=1}^{n_{\text{node}}} \int_{\mathcal{B}} \text{GRAD}\ N_i \cdot \hat{\mathbf{S}}^h(\tilde{\mathbf{C}}^h) \text{GRAD}\ N_j dV \tilde{\mathbf{q}}^j \end{aligned} \tag{7.189}$$

Notice the main difference from Algorithms 17 and 18 is that we discretize the discrete right Cauchy-Green deformation tensor independently from the nodal configuration vector. Hence, the algorithmic discrete right Cauchy-Green deformation tensor cannot be directly defined through the algorithmic deformation gradient tensor field, i.e.,

$$\tilde{\mathbf{C}}^h \neq \tilde{\mathbf{F}}^{h\,T} \tilde{\mathbf{F}}^h \tag{7.190}$$

3. *When selecting* $U0(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\text{s}}) = U0(1,1,1)$ *or* $V0(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\text{s}}) = V0(1,1,\rho_\infty^{\text{s}})$ *with* $\rho_\infty^{\text{s}} \in [0,1]$*, we recover the* ***assumed strain method*** *[59] in the sense of the three-root system, and it conserves the discrete mechanical energy in* $[t_n, t_{n+1}]$ *in the sense of* $\mathcal{E}_{n+1}^h = \mathcal{E}_n^h$ *only for the St. Venant-Kirchhoff material models. For a general hyperelastic material model such as Neo-Hookean or Ogden nonlinear material models, the discrete mechanical energy is not strictly conserved (see modified Option III for extensions).*

4. *The discrete total angular momentum in* $[t_n, t_{n+1}]$ *is exactly conserved under the same conditions as that for the symplectic members; that is,* $U0(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\text{s}}) = U0(1,1,1)$ *or* $V0(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\text{s}}) = V0(1,1,\rho_\infty^{\text{s}})$ *with* $\rho_\infty^{\text{s}} \in [0,1]$*. The discrete total linear momentum within the time step is exactly conserved for any conditions. The discrete total energy is not conserved exactly, but is bounded.*

From a comparison with the second-order time accurate energy-momentum conserving algorithm by Simo and Gonzalez [60] and Gonzales [61] in the single-field form, Algorithm 26 may be re-written as follows for general hyperelastic material models:

Algorithm 27

I-GSSSS Algorithms for Nonlinear Elastodynamics: Modified Option III (Energy-Momentum Conserving/Dissipative Methods)

Integrator:

$$\sum_{j=1}^{n_{\text{node}}} M_{ij}\tilde{\mathbf{a}} + \int_{\mathcal{B}} \tilde{\mathbf{F}}^h \cdot \mathbf{S}^h_{\text{alg}} \,\text{GRAD}\, N_i \, dV = \mathbf{F}_i^{\text{ext}^h}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}, t_{n+W_1})$$

with the (modified) algorithmic second Piola-Kirchhoff stress tensor $\mathbf{S}^h_{\text{alg}}$ *defined as*

$$\mathbf{S}^h_{\text{alg}} := 2D\hat{W}(\tilde{\mathbf{C}}^h) + 2\chi \frac{\hat{W}(\mathbf{C}^h_{n+1}) - \hat{W}(\mathbf{C}^h_n) - D\hat{W}(\tilde{\mathbf{C}}^h) : \Delta\mathbf{C}^h}{\| \Delta\mathbf{C}^h \|^2} \Delta\mathbf{C}^h$$

where $\chi := 1 - \text{sgn}^2(W_1 - \frac{1}{2})$*, and the algorithmic unknowns are*

$$\begin{aligned}
\tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1\Lambda_6\Delta\mathbf{a} \\
\tilde{\mathbf{v}} &= \dot{\mathbf{q}}_n + W_1\Lambda_4\ddot{\mathbf{q}}_n\Delta t + W_2\Lambda_5\Delta\mathbf{a}\Delta t \\
\tilde{\mathbf{q}} &= \mathbf{q}_n + W_1\Lambda_1\dot{\mathbf{q}}_n\Delta t + W_2\Lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + W_3\Lambda_3\Delta\mathbf{a}\Delta t^2 \\
\tilde{\mathbf{C}}^h &= \mathbf{C}^h_n + W_3(\mathbf{C}^h_{n+1} - \mathbf{C}^h_n) + (W_1 - W_3)\dot{\mathbf{C}}^h_n\Delta t
\end{aligned}$$

Updates:

$$\begin{aligned}
\mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + \lambda_3\Delta\mathbf{a}\Delta t^2 \\
\dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5\Delta\mathbf{a}\Delta t \\
\ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_n + \Delta\mathbf{a}
\end{aligned}$$

Initial conditions:

$$\begin{aligned}
\mathbf{q}(t_0) &= \mathbf{q}_0 \\
\dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0
\end{aligned}$$

Algorithmic parameters:

U0 Based-Option III Algorithms: *Employ the U0-family of algorithmic parameters from Algorithm 2*

V0 Based-Option III Algorithms: *Employ the V0-family of algorithmic parameters from Algorithm 3*

Remark 26 (Algorithm 27)

1. *The algorithmic parameters are directly employed from Algorithms 2 and 3 for the U0-family and V0-family based schemes, respectively.*

2. *Algorithm 27 recovers the exact energy-momentum conserving algorithm and the underlying features described by Simo and Gonzalez [60] or Gonzales [61] for* $U0(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}}) = U0(1,1,1)$ *or* $V0(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}}) = V0(1,1,\rho_\infty^{\mathrm{s}})$ *with* $\rho_\infty^{\mathrm{s}} \in [0,1]$ *in the sense of the single-field form/three-root (with two principal roots and a spurious root) algorithm design. The algorithmic right Cauchy-Green deformation tensor at* t_n *and* t_{n+1} *is evaluated by* $\mathbf{C}_n^h = \mathbf{C}^h(\mathbf{q}_n)$ *and* $\mathbf{C}_{n+1}^h = \mathbf{C}^h(\mathbf{q}_{n+1})$*, respectively, and we define* $\Delta\mathbf{C}^h := \mathbf{C}_{n+1}^h - \mathbf{C}_n^h$ *and* $\| \Delta\mathbf{C}^h \|^2 := \Delta\mathbf{C}^h : \Delta\mathbf{C}^h$*. The function* χ *defined with the sign function is given as*

$$\begin{aligned}\chi &:= 1 - \mathrm{sgn}^2(W_1 - \tfrac{1}{2}) = 4H_{1/2}(W_1 - \tfrac{1}{2})\left[1 - H_{1/2}(W_1 - \tfrac{1}{2})\right] \\ &= \begin{cases} 1 & \text{for } W_1 = \frac{1}{2} \\ 0 & \text{for } W_1 \neq \frac{1}{2} \end{cases}\end{aligned} \tag{7.191}$$

where $H_{1/2}$ *is the Heaviside step function. Therefore, the algorithmic second Piola-Kirchhoff stress tensor* $\mathbf{S}_{\mathrm{alg}}^h$ *is given as*

$$\mathbf{S}_{\mathrm{alg}}^h = 2D\hat{W}(\tilde{\mathbf{C}}^h) + 2\frac{\hat{W}(\mathbf{C}_{n+1}^h) - \hat{W}(\mathbf{C}_n^h) - D\hat{W}(\tilde{\mathbf{C}}^h) : \Delta\mathbf{C}^h}{\| \Delta\mathbf{C}^h \|^2}\Delta\mathbf{C}^h \tag{7.192}$$

for $W_1 = \frac{1}{2}$ *and*

$$\mathbf{S}_{\mathrm{alg}}^h = 2D\hat{W}(\tilde{\mathbf{C}}^h) \quad \text{for } W_1 \neq \tfrac{1}{2} \tag{7.193}$$

Note that, for the St. Venant-Kirchhoff material model, Eq. (7.192) *can be reduced to* $\mathbf{S}_{\mathrm{alg}}^h = 2D\hat{W}(\mathbf{C}_{n+1/2}^h)$*. The (mechanical) energy change is*

defined as, $\mid \mathscr{E}_0 - \mathscr{E}_n \mid$ *where* $\mathscr{E}_0$ *and* $\mathscr{E}_n$ *denote the discrete mechanical energy at the initial time and the current time* t_n*, respectively. When* $U0/V0(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}}) = U0/V0(1,1,1)$ *or* $V0(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}}) = V0(1,1,0)$ *is selected, the mechanical energy is exactly conserved. Notice that the configuration, velocity, and acceleration are second-order time convergent.*

3. *Since* $\bar{\mathbf{S}}^h := 2\frac{\hat{W}(\mathbf{C}_{n+1}^h) - \hat{W}(\mathbf{C}_n^h) - D\hat{W}(\tilde{\mathbf{C}}^h):\Delta\mathbf{C}^h}{\left\|\Delta\mathbf{C}^h\right\|^2}\Delta\mathbf{C}^h = \mathcal{O}(\Delta t^2)$ *and the algorithmic time level of the evaluation of the Lagrange strain tensor is at* $t \approx t_{n+1/2}$*, it is appropriate to recover the energy-momentum conserving algorithm under the conditions of* $U0(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}}) = U0(1,1,1)$ *or* $V0(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}}) = V0(1,1,\rho_\infty^{\mathrm{s}})$ *with* $\rho_\infty^{\mathrm{s}} \in [0,1]$*. It is important to note that the algorithmic time level of the algorithmic second Piola-Kirchhoff stress tensor coincides with the algorithmic time level of the symplectic-momentum conserving schemes and the assumed strain method in the single-field form/three-root system in Algorithm 26.*

7.3 Numerical Results

In this section, various numerical results are shown with the time step size of $\Delta t = 0.01$ sec. To demonstrate and illustrate the numerical behavior of the implicit family of algorithms presented in this chapter, we again consider the following three problems: (1) Nonlinear oscillator problem [1], (2) (Classical) Kepler's problem, and (3) Lennard-Jones (5, 3) Potential 2-body Problem. The information of the input parameters and such are the same for the simulations presented in Section 6.3. The brief summaries of the numerical examples are shown in Appendix A. We demonstrate the implicit family of algorithms in the sense of Option I and Option II. The selected algorithms are:
Implicit GSSSS family of algorithms (Option III): Algorithm 22
Implicit GSSSS family of algorithms (Modified Option III): Algorithm 23
For each algorithm, we select,

U0V0/V0U0(ρ_∞,1.0,ρ_∞)
U0V0(1.0,1.0,$\rho_\infty^{\rm s}$)
V0U0(1.0,1.0,$\rho_\infty^{\rm s}$)
where

$$\begin{aligned}\rho_\infty &= \rho_\infty^{\min} = \rho_\infty^{\max} \in \{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\} \\ \rho_\infty^{\rm s} &\in \{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\}\end{aligned} \tag{7.194}$$

The legends of the figures are shown in Fig. 4.5.

Nonlinear oscillator problem: The second-order time accuracies in the configuration, velocity, and acceleration in the forced-damped system, Eq. (A.31) with $C = 0.001$ and $\mathbf{f}^{\rm ext} = \bar{\mathbf{f}}\sin(t)$ where $\bar{\mathbf{f}} = (0.1, -0.5)^T$, with the selected schemes from the family of implicit GSSSS algorithms in the sense of Option III and the modified Option III are shown in Fig. 7.1 and Fig. 7.2, respectively. The configuration, velocity, and acceleration have been used to calculate the reference errors,

$$\begin{aligned}{\rm error}_q &= \left| \frac{q^1 - q^1_{\rm ref}}{q^1_{\rm ref}} \right| \\ {\rm error}_v &= \left| \frac{\dot{q}^1 - \dot{q}^1_{\rm ref}}{\dot{q}^1_{\rm ref}} \right| \\ {\rm error}_a &= \left| \frac{\ddot{q}^1 - \ddot{q}^1_{\rm ref}}{\ddot{q}^1_{\rm ref}} \right|\end{aligned} \tag{7.195}$$

respectively, at time $t = 1$ sec. The reference configurations, velocities, and accelerations have been obtained with a sufficiently small time step size $\Delta t = 10^{-5}$. As can be seen from the figures, all time integration schemes are second-order time accurate in all the configurations, velocities, and accelerations. It should be noted that the second-order time accuracies of the schemes not only in the configurations and velocities, but also the accelerations have been achieved from the consistency of the time level of the discrete algorithmic balance equations at time level $t = t_{n+W_1}$.

In this problem, the total linear momentum within a time step should not be

conserved; all schemes never show that $\mathbf{L}_n = \mathbf{L}_{n+1}$ as can be seen from Fig. 7.3 - Fig. 7.14, it is always bounded for any schemes in the family of algorithms in Option I and Option II both in the conservative and dissipative systems.

The conservation of the angular momentum in the sense of $\mathbf{J}_n = \mathbf{J}_{n+1}$ in the conservative system is achieved only for some special cases similar to the cases in Section 6.3. That is, only the particular family of schemes with $\rho_\infty^{\min} = \rho_\infty^{\max} = \rho_\infty^{\mathrm{s}} = 1$ in the U0-based family or $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\mathrm{s}} \in [0, 1]$ in the V0-based family in both Option III and the modified Option III can conserve the angular momentum within the time step exactly. Fig. 7.3 and Fig. 7.6 illustrate that the U0V0/V0U0 optimal schemes in both Option III and the modified Option III are not angular momentum conserving schemes [except (1,1,1) case in both Option III and the modified Option III]; however, they give better results in the qualitative sense by selecting a high value of ρ_∞ close to but less than unity. In the dissipative system, all schemes show that the angular momentum is always decaying as expected.

As can be seen from Fig. 7.6 - Fig. 7.8, the mechanical energy of the system is exactly conserved in the sense of $\mathscr{E}_n = \mathscr{E}_{n+1}$ in the conservative system for $\rho_\infty^{\min} = \rho_\infty^{\max} = \rho_\infty^{\mathrm{s}} = 1$ in the U0-based family/$\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\mathrm{s}} \in [0, 1]$ in the V0-based family only in the modified Option III algorithms. In the dissipative system, all schemes selected show the dissipative behaviors of the mechanical energy. The U0V0$(1, 1, \rho_\infty^{\mathrm{s}})$ schemes in both Option III and the modified Option III show that the amplitude of the oscillation of the mechanical energy can be decreased by selecting a high value of ρ_∞^{s} which is less than unity. The U0V0/V0U0 optimal schemes show that the dissipative behavior of the mechanical energy in the conservative system tends to become lower more rapidly for a smaller ρ_∞. The initial conditions used for the simulation were:

$$\begin{aligned} \mathbf{q}(t_0) &= \mathbf{q}_0 = (0.1, 1.0)^T \\ \dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0 = (0.1, 0.1)^T \end{aligned} \tag{7.196}$$

The final time of the simulation is selected to be $T = 30$ sec.

3D Kepler's Problem: In this example problem, we again applied the same selected schemes from the previous example problem. And we observed similar numerical behaviors of the algorithms in the time accuracies and the linear and angular momenta and mechanical energy conservations. Similar to the nonlinear oscillator problem presented previously, the linear momentum should not be conserved in the absence of the external loadings. Fig. 7.15 and Fig. 7.16 show that the second-order time accuracies in the configuration, velocity, and acceleration in the forced-damped system, with $C = 0.01$ and the external loading $\mathbf{f}^{\text{ext}} = \bar{\mathbf{f}} \sin(t)$ where $\bar{\mathbf{f}} = (0.1, 0.0, 0.0)^T$, with the selected schemes from the family of implicit GSSSS algorithms in the sense of Option III and modified Option III are obtained. The time histories of the mechanical energy, linear and angular momenta, configuration, velocity, and acceleration in the conservative system are shown in Fig. 7.17 - Fig. 7.22. The initial conditions used for the simulation were:

$$\begin{aligned} \mathbf{q}(t_0) &= \mathbf{q}_0 = (0.9/\sqrt{2}, 0.0, 0.9/\sqrt{2})^T \\ \dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0 = (0.0, -100/9, 0.0)^T \end{aligned} \tag{7.197}$$

The final time of the simulation is selected to be $T = 20$ sec.

Lennard-Jones potential problem: Again, in this example problem, we applied the same selected schemes from the previous example problem. And we basically observed the same numerical behaviors of the algorithms in the time accuracies and the angular momenta and mechanical energy conservations. But, the total linear momentum should be conserved exactly in the sense of $\mathbf{L}_n = \mathbf{L}_{n+1}$ in this problem in the absence of the external loading. Fig. 7.25 - Fig. 7.30 show that the total linear momentum is exactly conserved in the conservative system for any choices of the spectral conditions in both Option III and modified Option III. Fig. 7.23 and Fig. 7.24 show that the second-order time accuracies are obtained in the configuration, velocity, and acceleration in the forced-damped system, with $C = 0.01$ and the external loading $\mathbf{f}^{\text{ext}} = \bar{\mathbf{f}} \sin(t)$ where $\bar{\mathbf{f}} = (0.1, 0.0, 0.0, 0.0, -0.5, 0.0)^T$, with the selected schemes from the family of implicit GSSSS algorithms in the sense of Option III and modified Option III. The

time histories of the mechanical energy, linear and angular momenta, configuration, velocity, and acceleration in the conservative system are shown in Fig. 7.25 - Fig. 7.30. The initial conditions used for the simulation were:

$$\begin{aligned} \mathbf{q}(t_0) &= \mathbf{q}_0 = (0.0, -0.5, 0.0, 0.0, 0.5, 0.0)^T \\ \dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0 = (5.0, 1.0, 2.0, 10.0, 3.0, -1.0)^T \end{aligned} \tag{7.198}$$

The final time of the simulation is selected to be $T = 2$ sec.
Here is the summary of the numerical results:

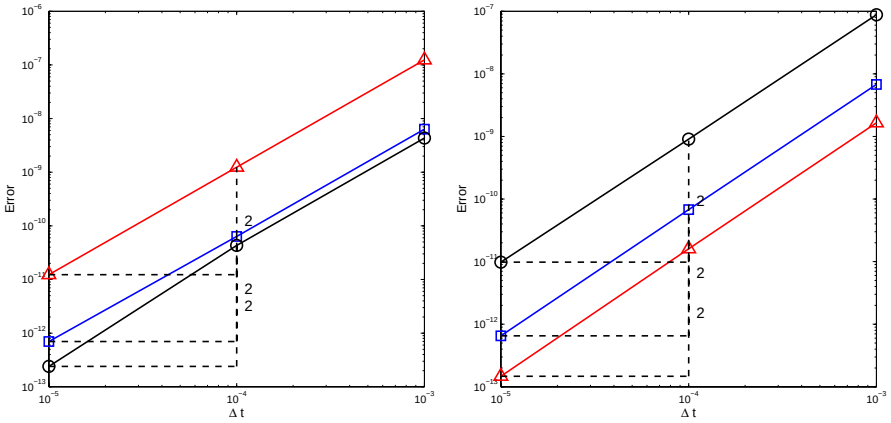
Remark 27

1. *The order of time accuracies in the configuration, velocity, and acceleration is 2 for any selections of the spectral conditions.*

2. *The energy conserving schemes are obtained by selecting the following spectral conditions in the conservative system:*
   Modified Option III: $\rho_\infty^{\min} = \rho_\infty^{\max} = \rho_\infty^{\mathrm{s}} = 1$ *in the U0-based family/*$\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ *and* $\rho_\infty^{\mathrm{s}} \in [0, 1]$ *in the V0-based family.*

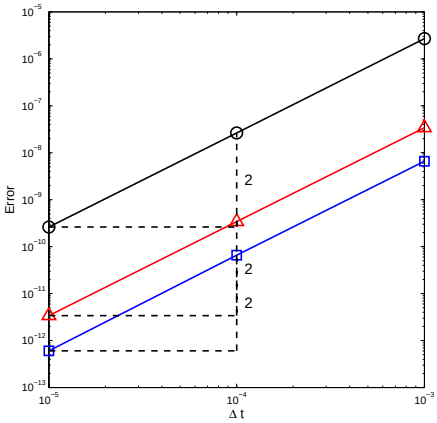
3. *The angular momentum conserving schemes are obtained by selecting the following spectral conditions in the conservative system:*
   Option III/Modified Option III: $\rho_\infty^{\min} = \rho_\infty^{\max} = \rho_\infty^{\mathrm{s}} = 1$ *in the U0-based family/*$\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ *and* $\rho_\infty^{\mathrm{s}} \in [0, 1]$ *in the V0-based family.*

4. *In the dissipative system, all schemes show the energy and angular momentum dissipative features.*

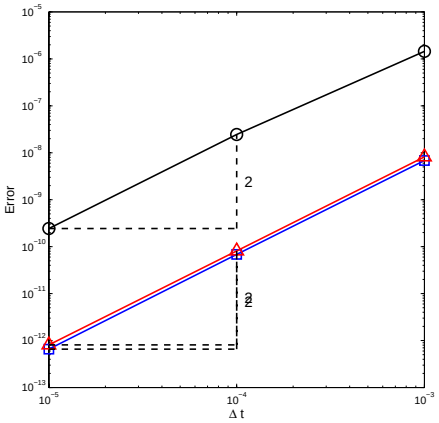


(a) U0V0/V0U0(1.0,1.0,1.0)

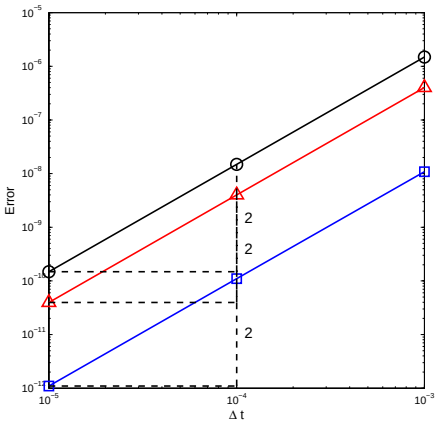
(b) U0V0(1.0,1.0,0.0)



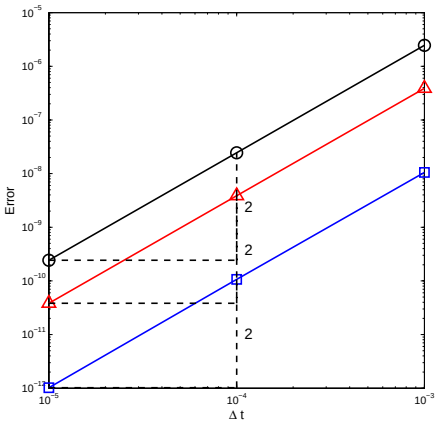
(c) V0U0(1.0,1.0,0.0)



(d) U0V0/V0U0(0.8,1.0,0.8)

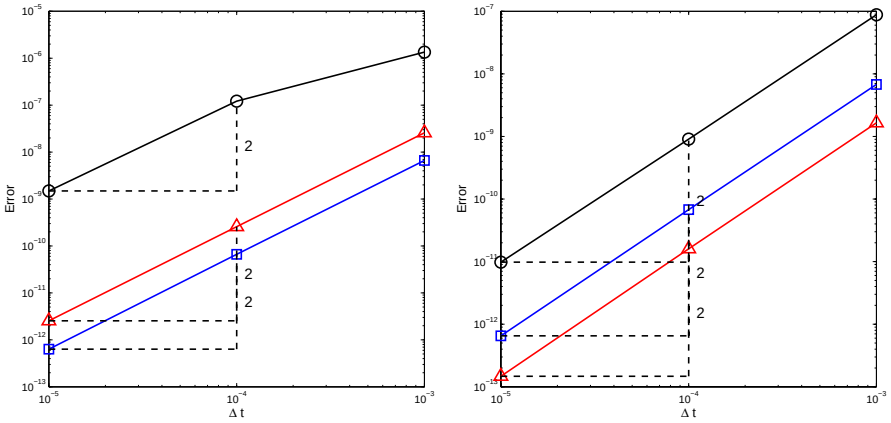


(e) U0V1(0.6,0.9,0.2)



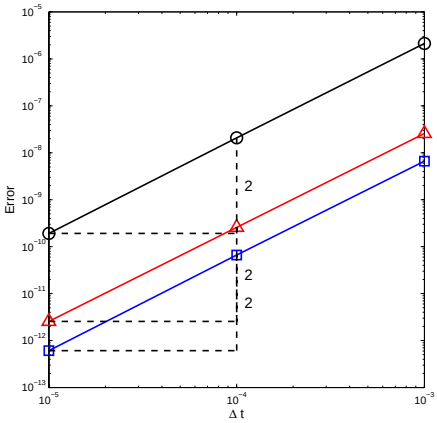
(f) V0U1(0.6,0.9,0.2)

Figure 7.1: Time accuracies in the configuration, velocity, and acceleration in the forced mechanical system. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Option III)]

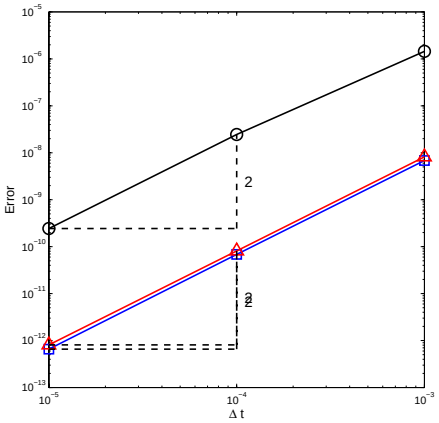


(a) U0V0/V0U0(1.0,1.0,1.0)

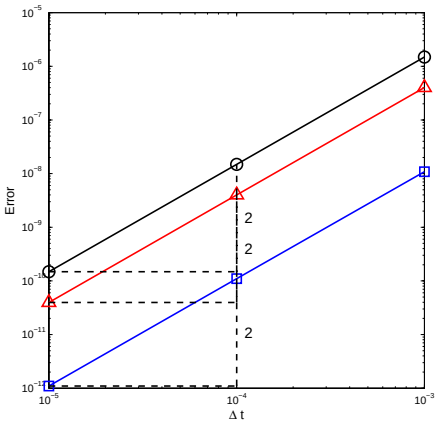
(b) U0V0(1.0,1.0,0.0)



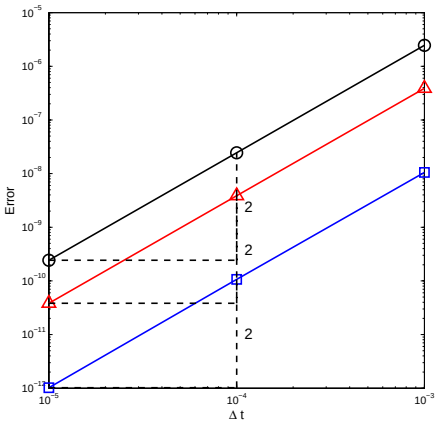
(c) V0U0(1.0,1.0,0.0)



(d) U0V0/V0U0(0.8,1.0,0.8)

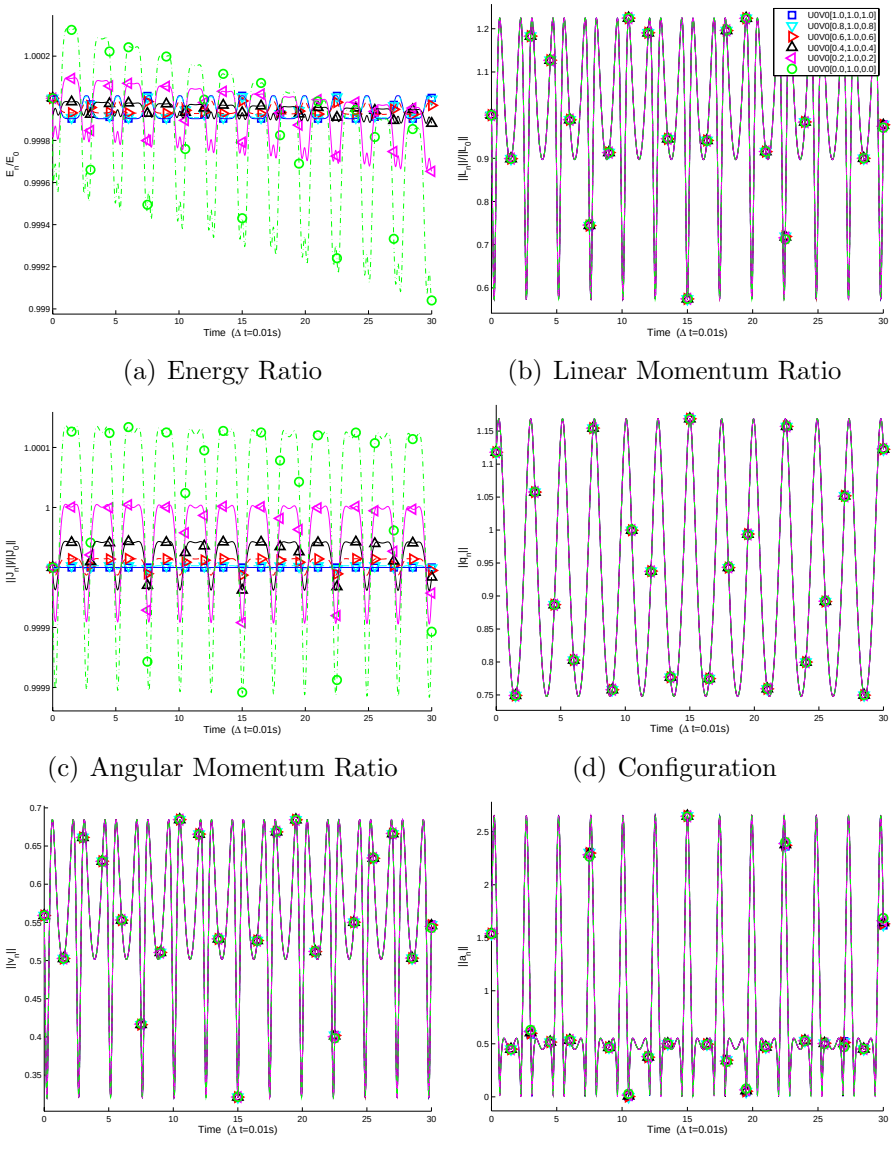


(e) U0V1(0.6,0.9,0.2)



(f) V0U1(0.6,0.9,0.2)

Figure 7.2: Time accuracies in the configuration, velocity, and acceleration in the forced mechanical system. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Modified Option III)]

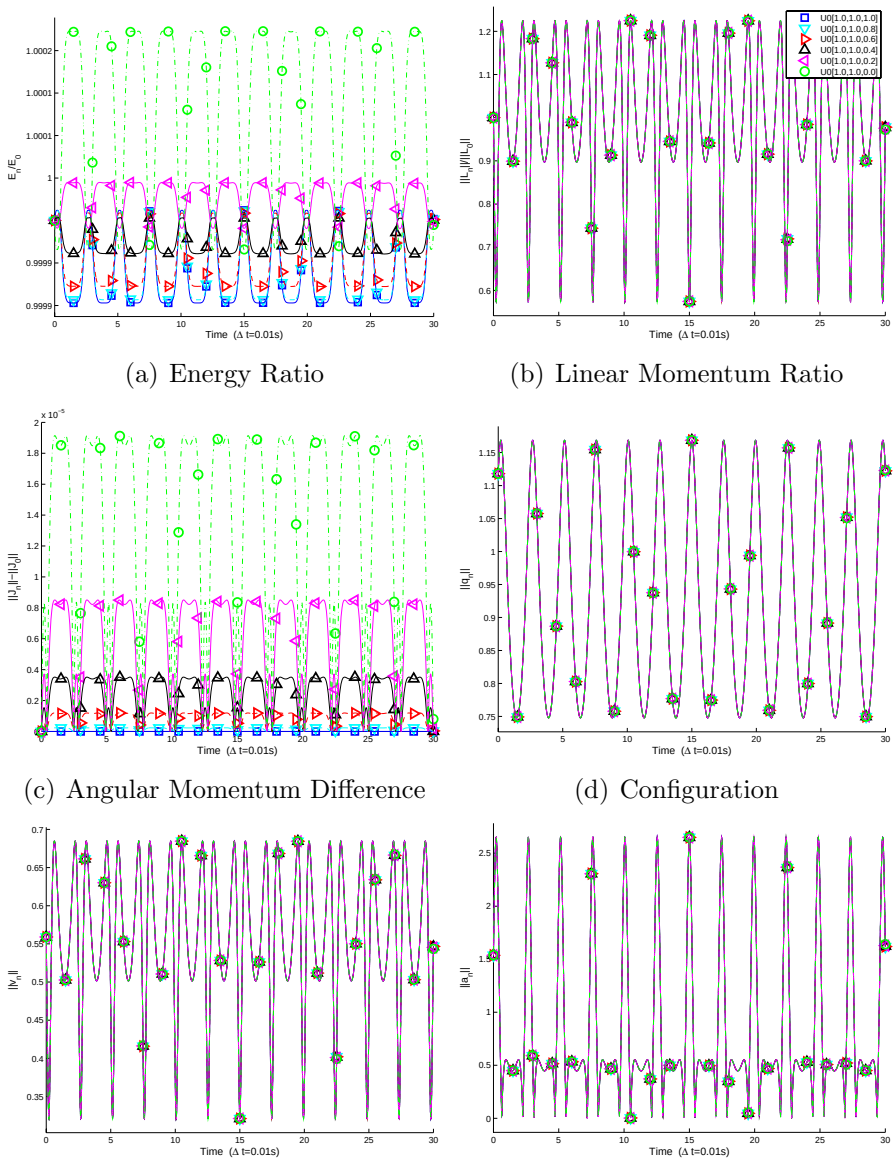


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 7.3: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Option III) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]

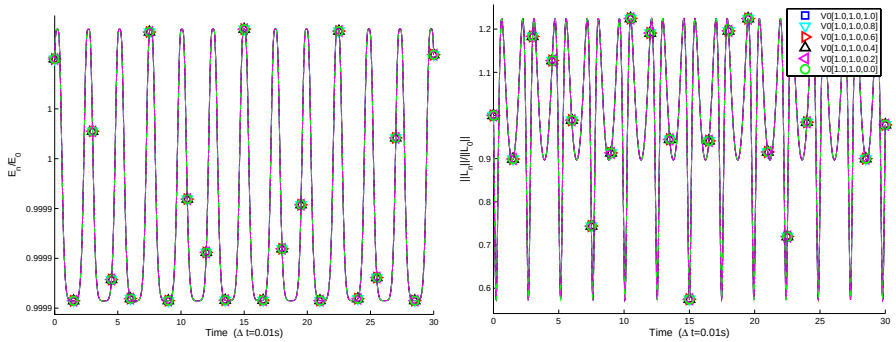


(a) Energy Ratio (b) Linear Momentum Ratio

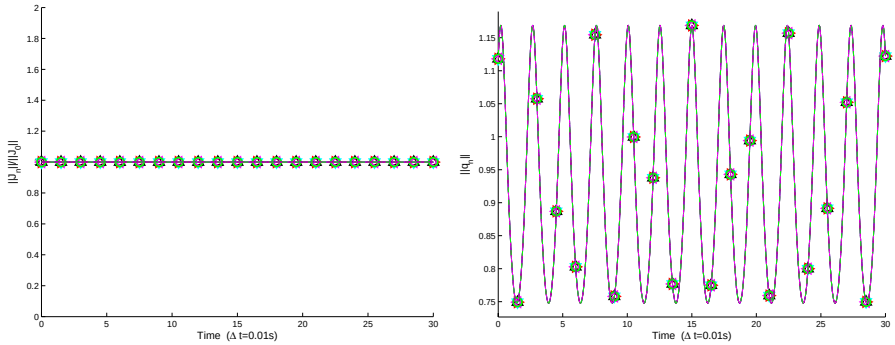
(c) Angular Momentum Difference (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

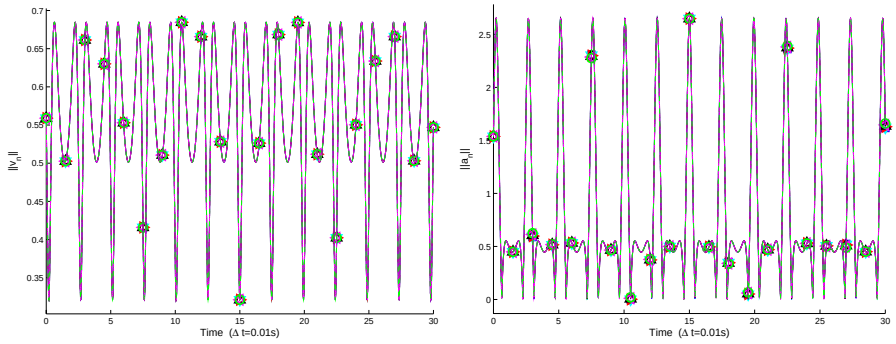
Figure 7.4: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Option III) - U0V0(1.0,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum Ratio

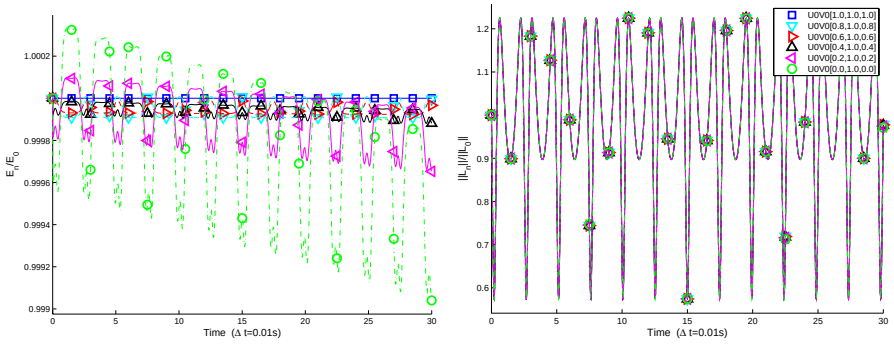


(c) Angular Momentum Ratio (d) Configuration

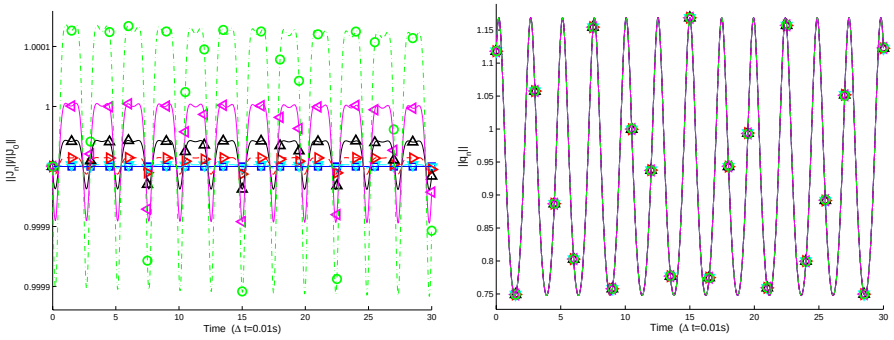


(e) Velocity Norm (f) Acceleration Norm

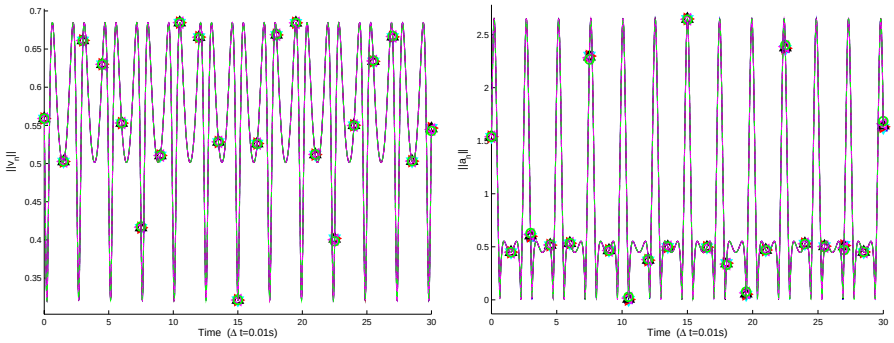
Figure 7.5: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Option III) - V0U0(1.0,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum Ratio

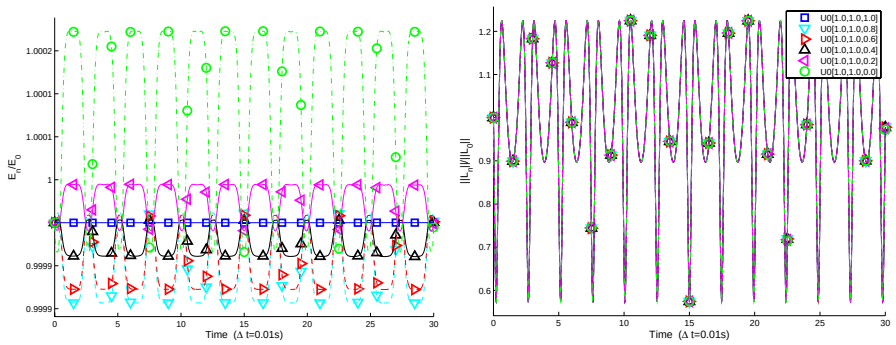


(c) Angular Momentum Ratio (d) Configuration

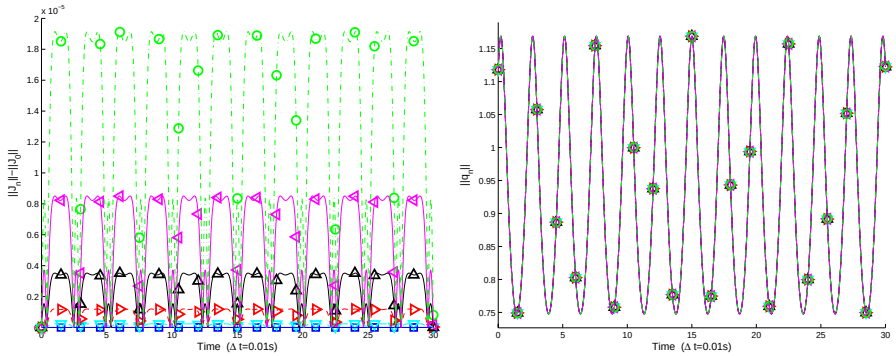


(e) Velocity Norm (f) Acceleration Norm

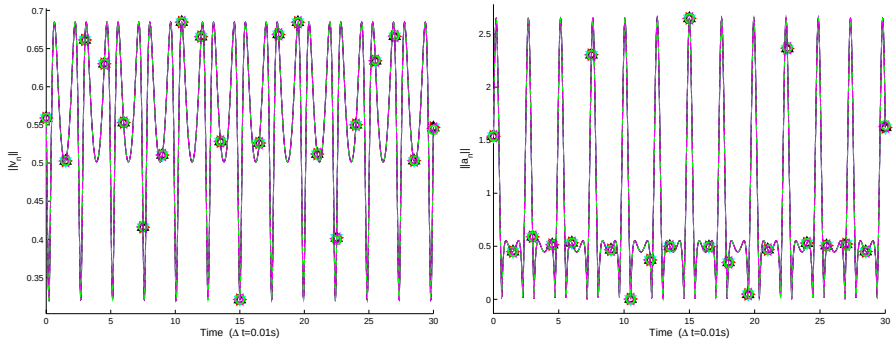
Figure 7.6: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Modified Option III) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum Ratio

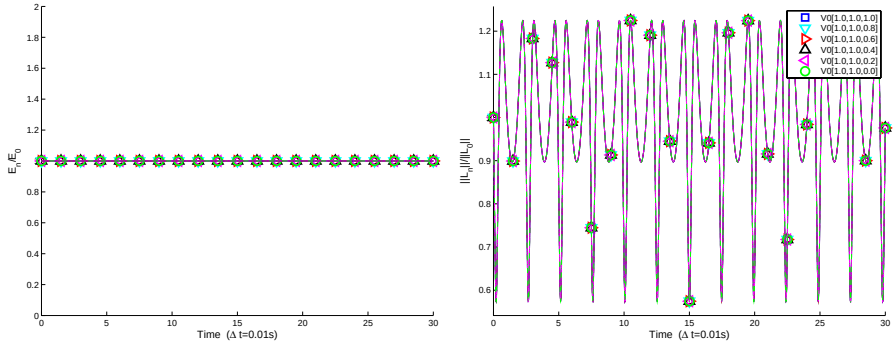


(c) Angular Momentum Difference (d) Configuration

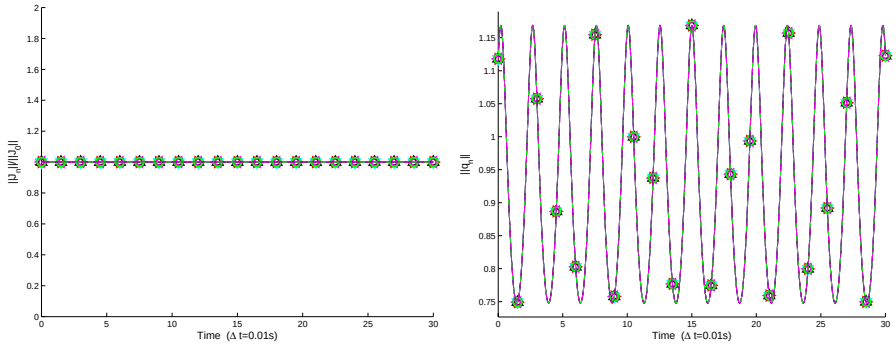


(e) Velocity Norm (f) Acceleration Norm

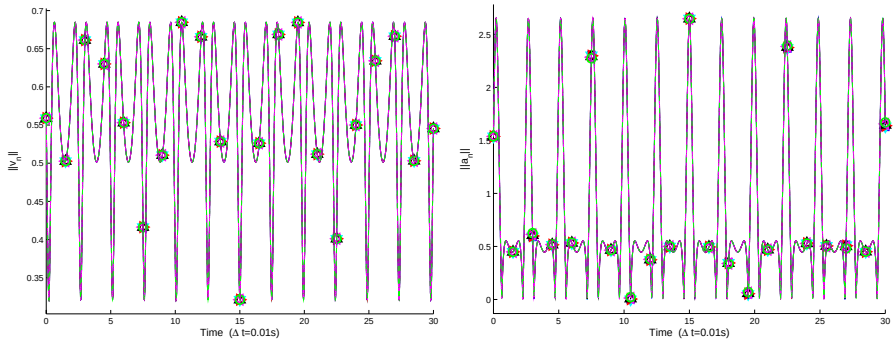
Figure 7.7: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Modified Option III) - U0V0(1.0,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum Ratio

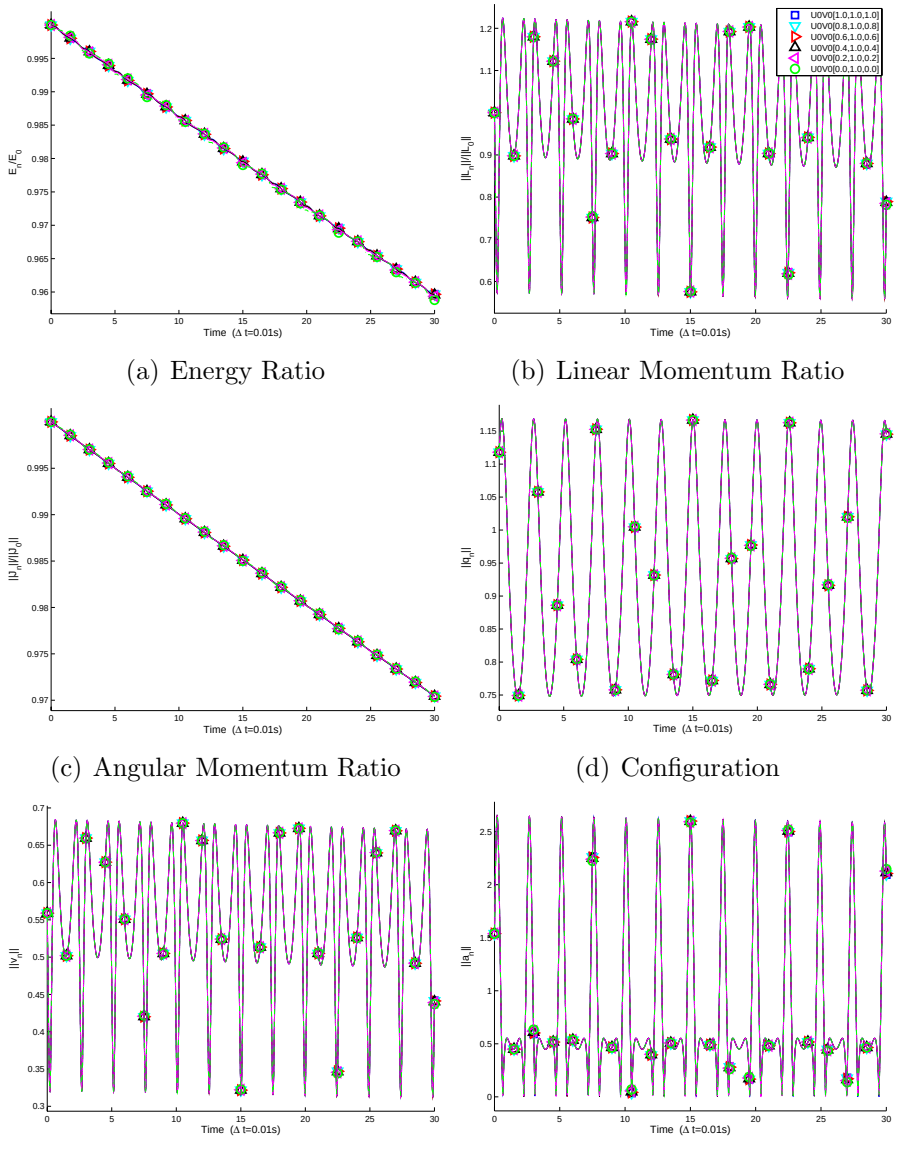


(c) Angular Momentum Ratio (d) Configuration



(e) Velocity Norm (f) Acceleration Norm

Figure 7.8: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Modified Option III) - V0U0(1.0,1.0,ρ_∞)]

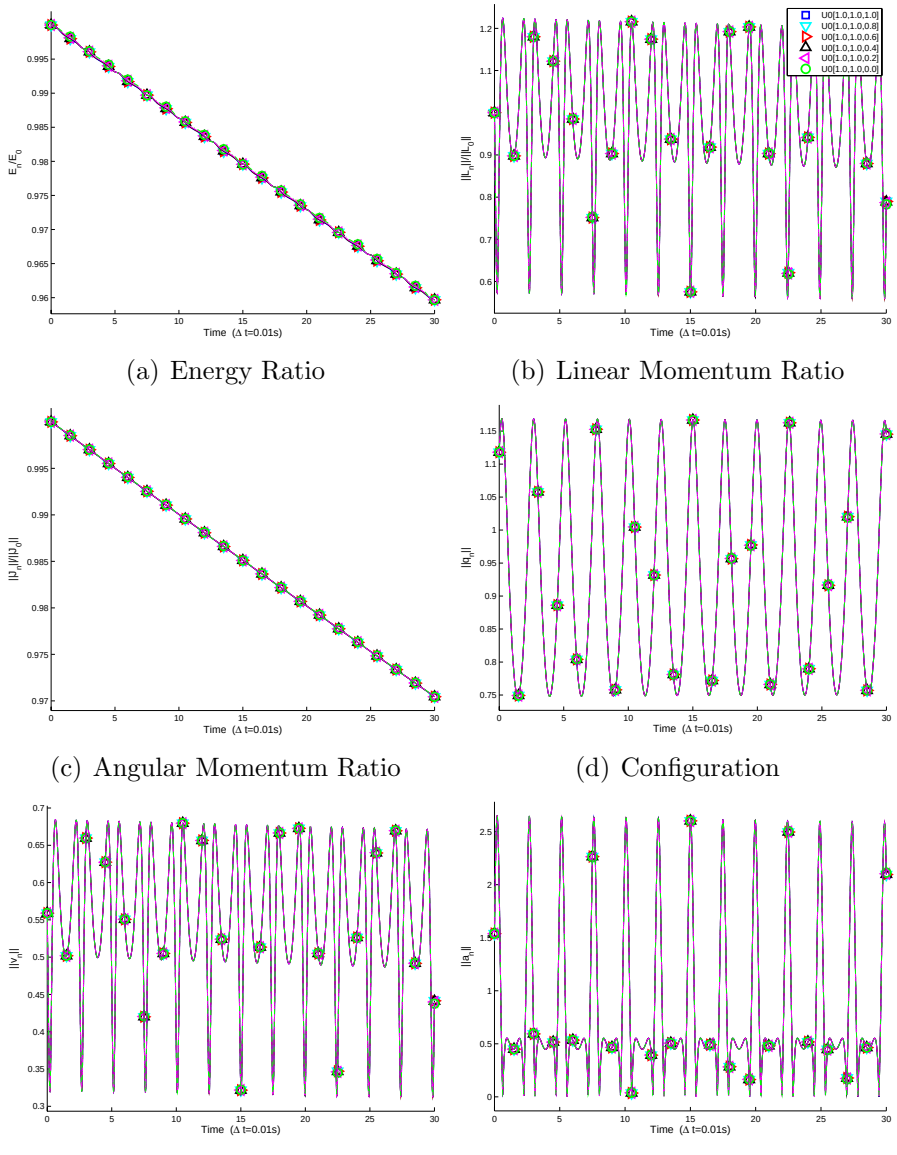


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 7.9: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Option III) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]

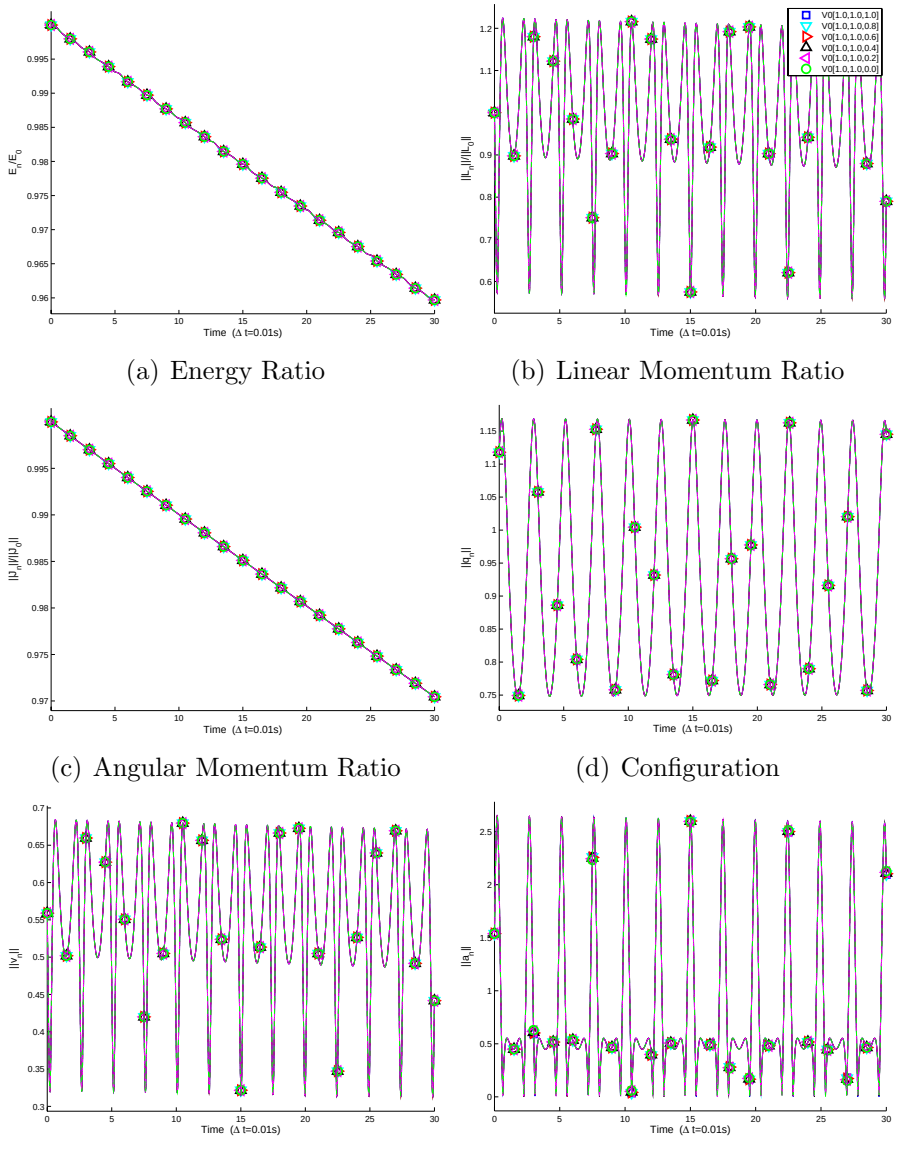


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 7.10: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Option III) - U0V0(1.0,1.0,ρ_∞)]

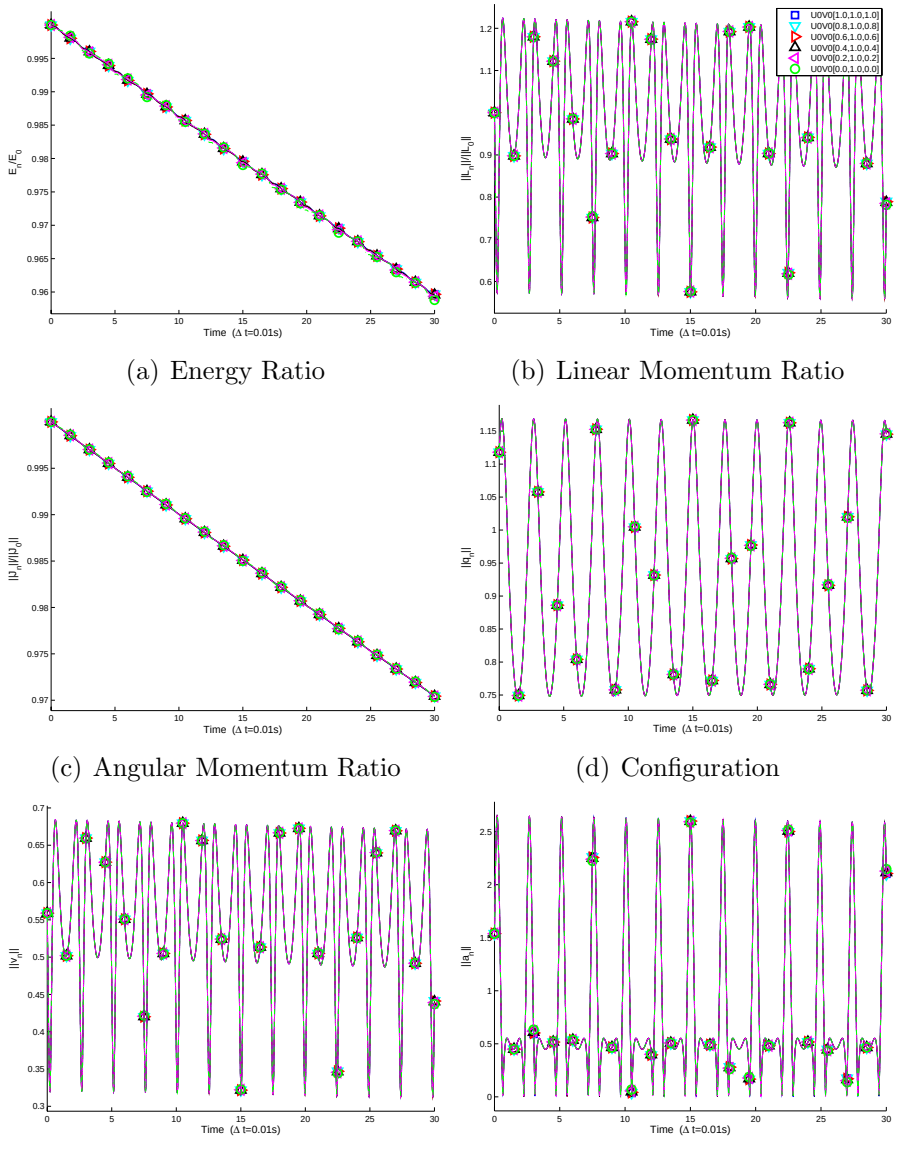


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 7.11: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Option III) - V0U0(1.0,1.0,ρ_∞)]

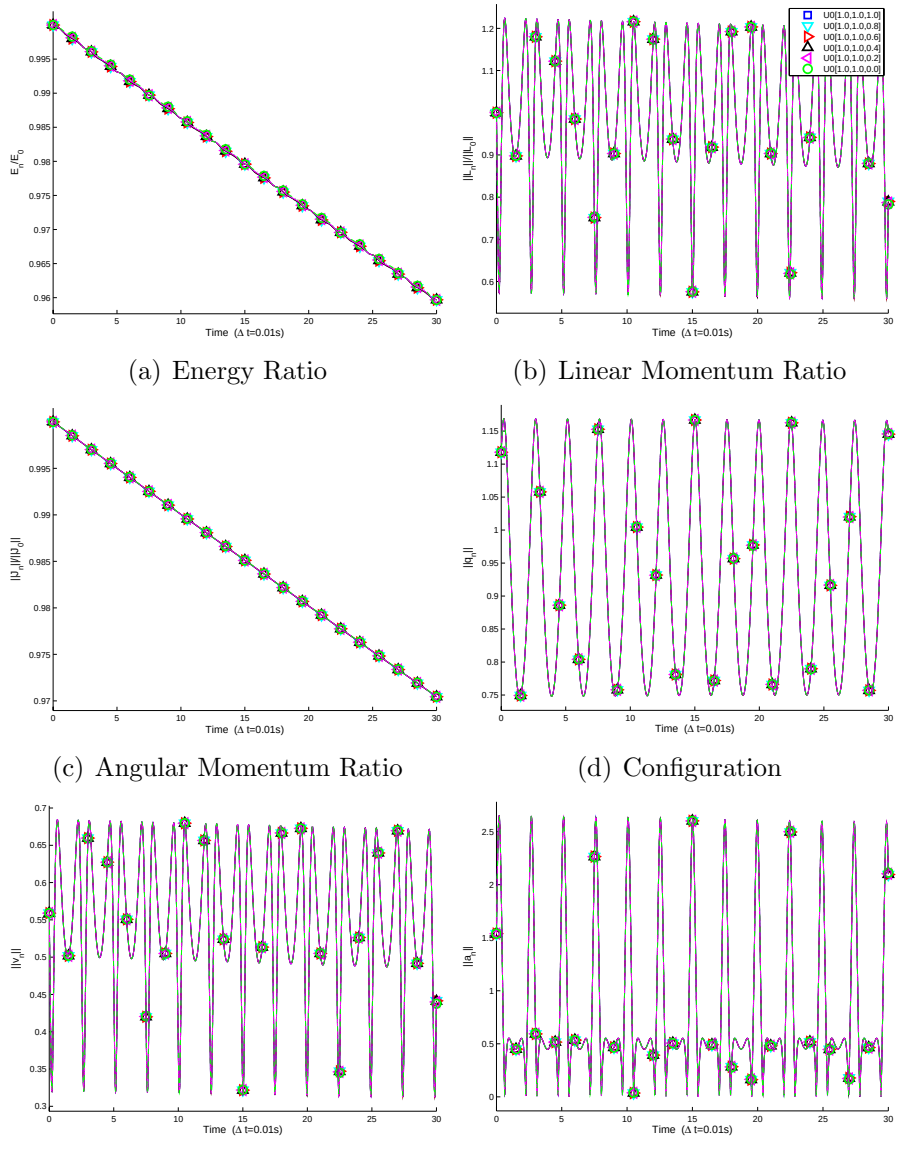


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 7.12: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Modified Option III) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]

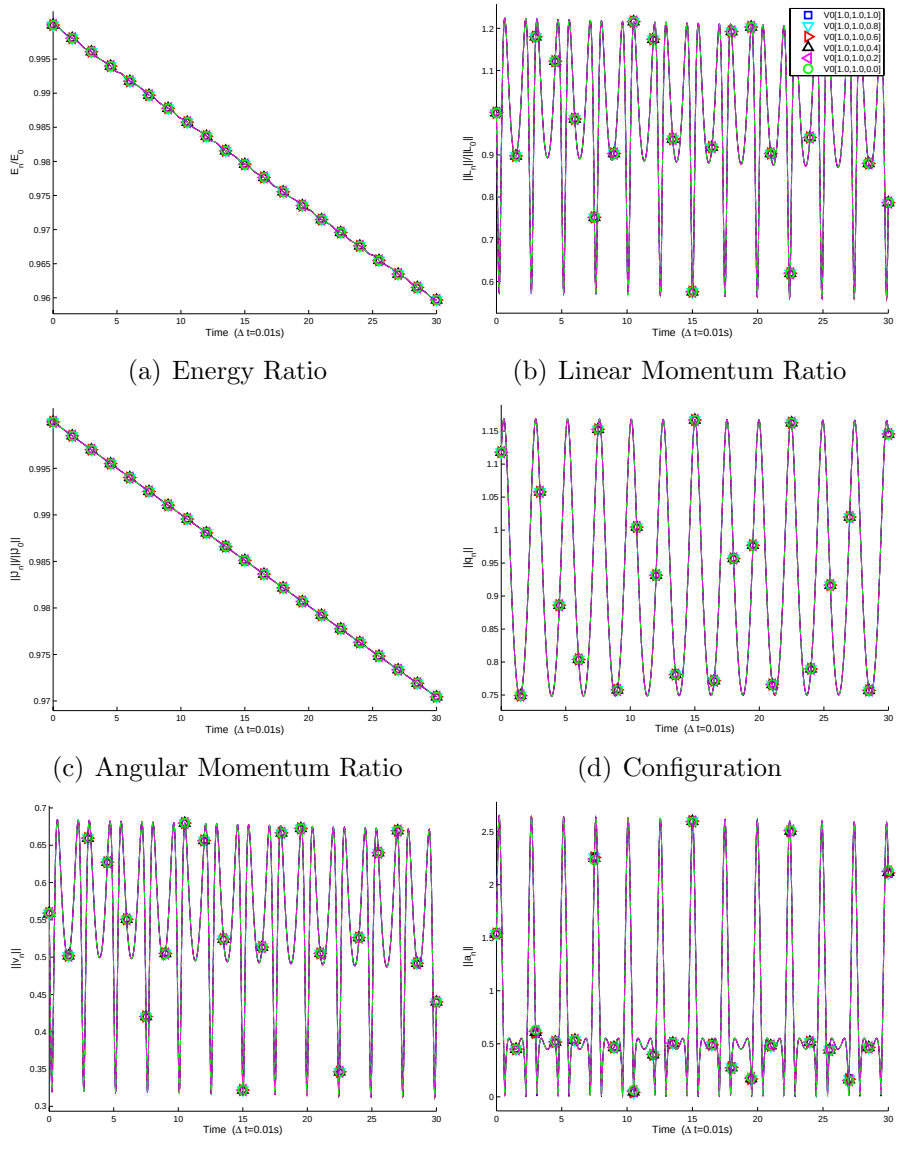


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 7.13: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Modified Option III) - U0V0(1.0,1.0,ρ_∞)]

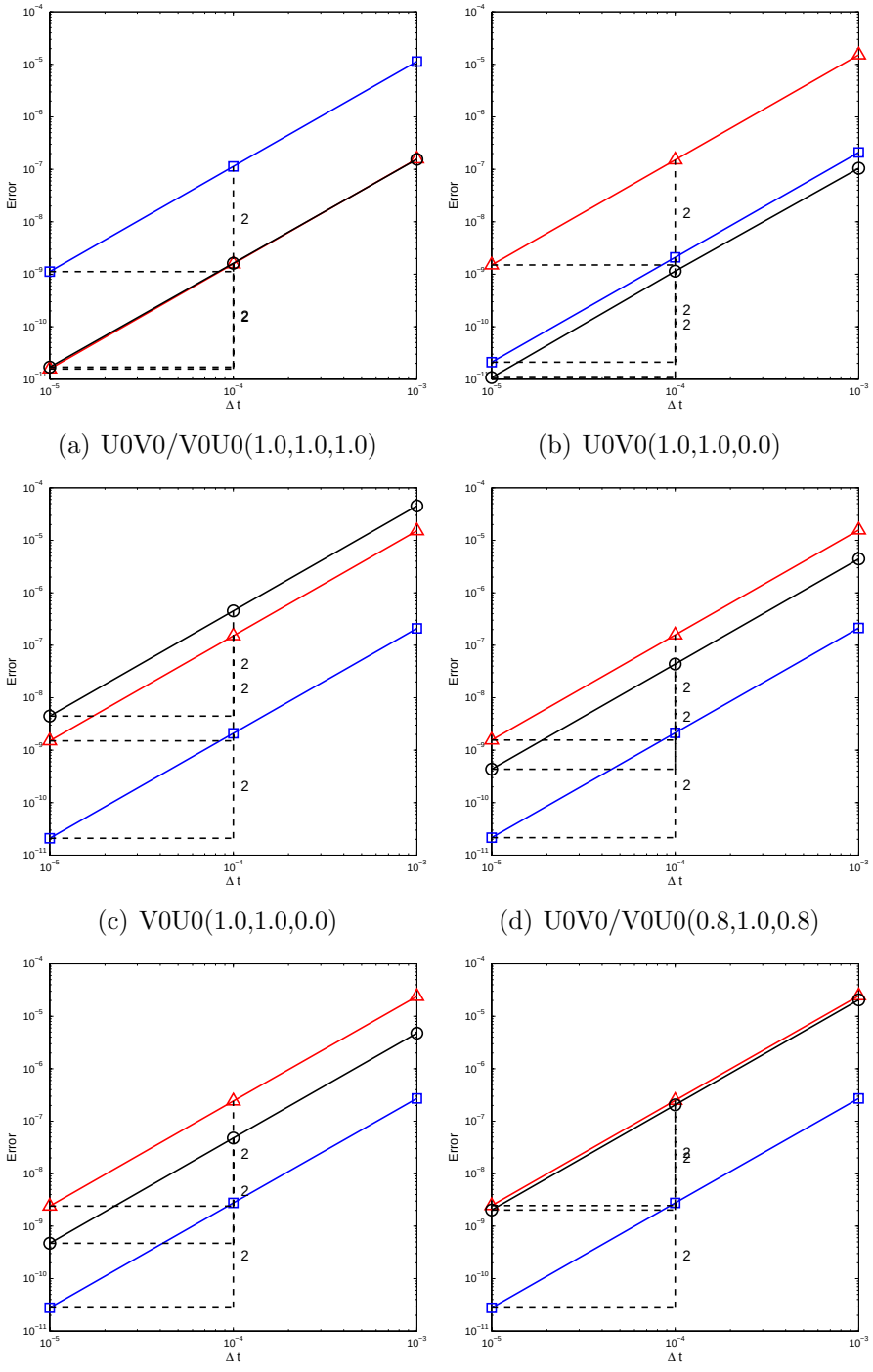


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 7.14: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Implicit GSSSS family of algorithms (Modified Option III) - V0U0(1.0,1.0,ρ_∞)]



(a) U0V0/V0U0(1.0,1.0,1.0)

(b) U0V0(1.0,1.0,0.0)

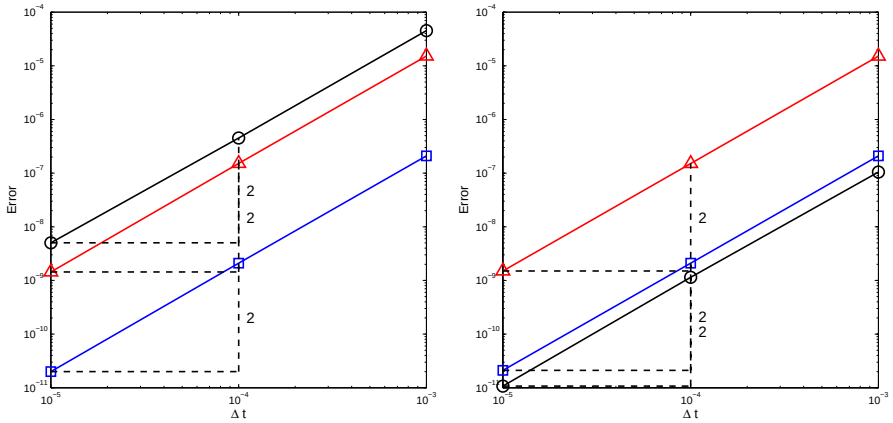
(c) V0U0(1.0,1.0,0.0)

(d) U0V0/V0U0(0.8,1.0,0.8)

(e) U0V1(0.6,0.9,0.2)

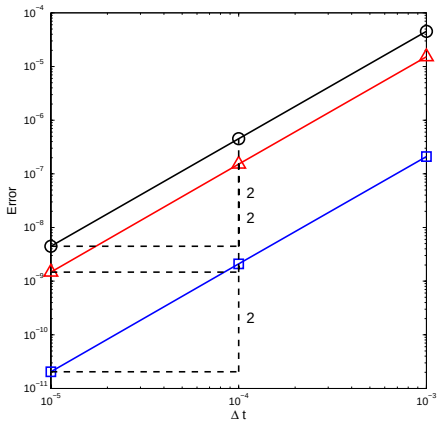
(f) V0U1(0.6,0.9,0.2)

Figure 7.15: Time accuracies in the configuration, velocity, and acceleration in the forced mechanical system. [Problem: Kepler's problem] [Algorithm: Implicit GSSSS family of algorithms (Option III)]

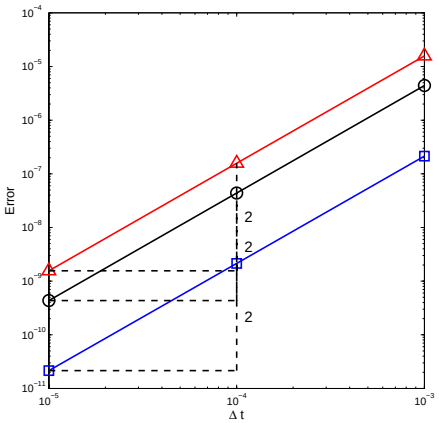


(a) U0V0/V0U0(1.0,1.0,1.0)

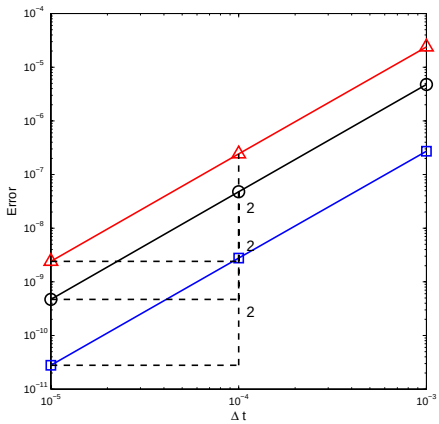
(b) U0V0(1.0,1.0,0.0)



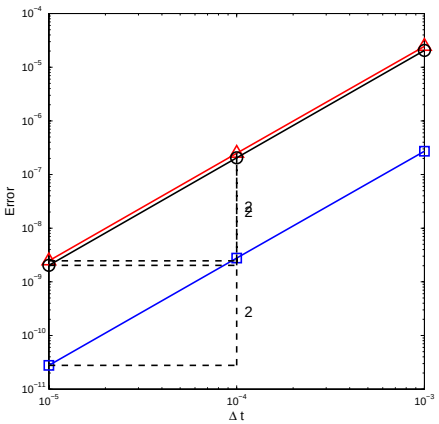
(c) V0U0(1.0,1.0,0.0)



(d) U0V0/V0U0(0.8,1.0,0.8)

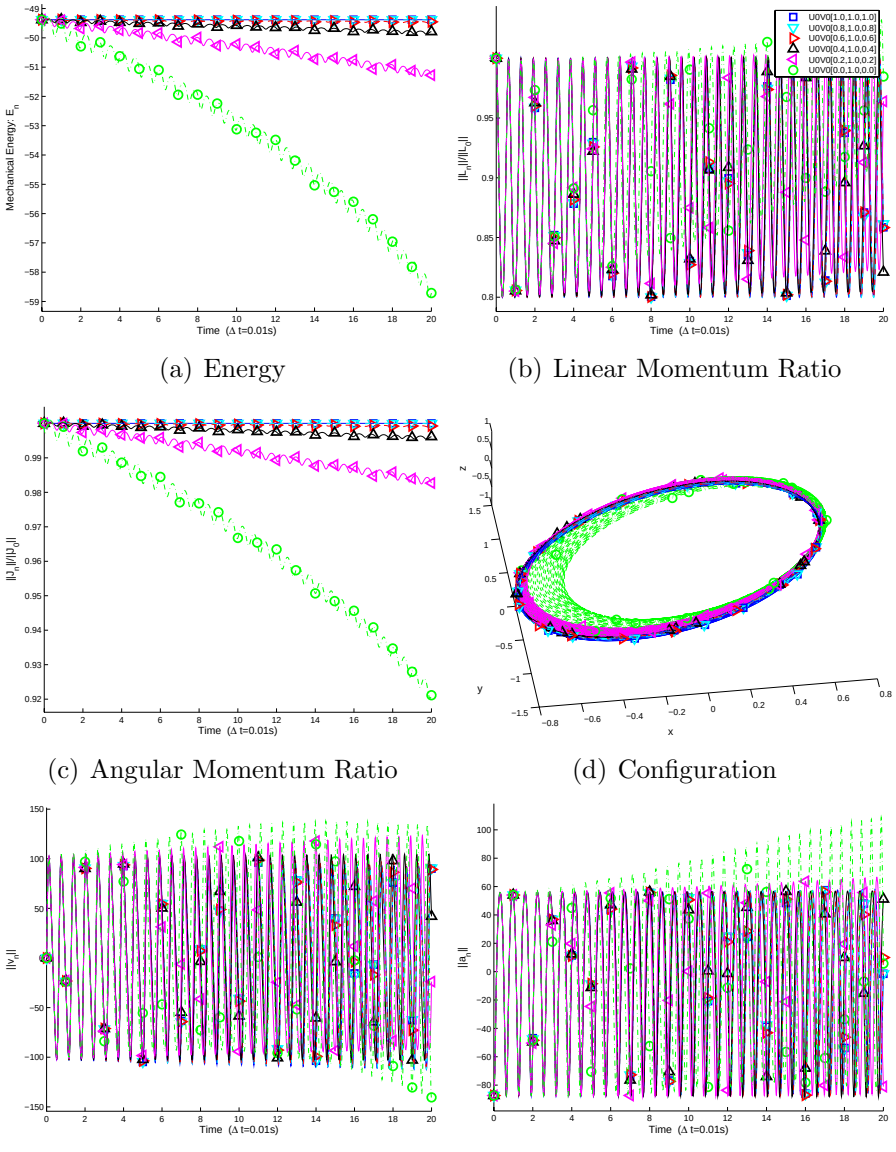


(e) U0V1(0.6,0.9,0.2)



(f) V0U1(0.6,0.9,0.2)

Figure 7.16: Time accuracies in the configuration, velocity, and acceleration in the forced mechanical system. [Problem: Kepler's problem] [Algorithm: Implicit GSSSS family of algorithms (Modified Option III)]

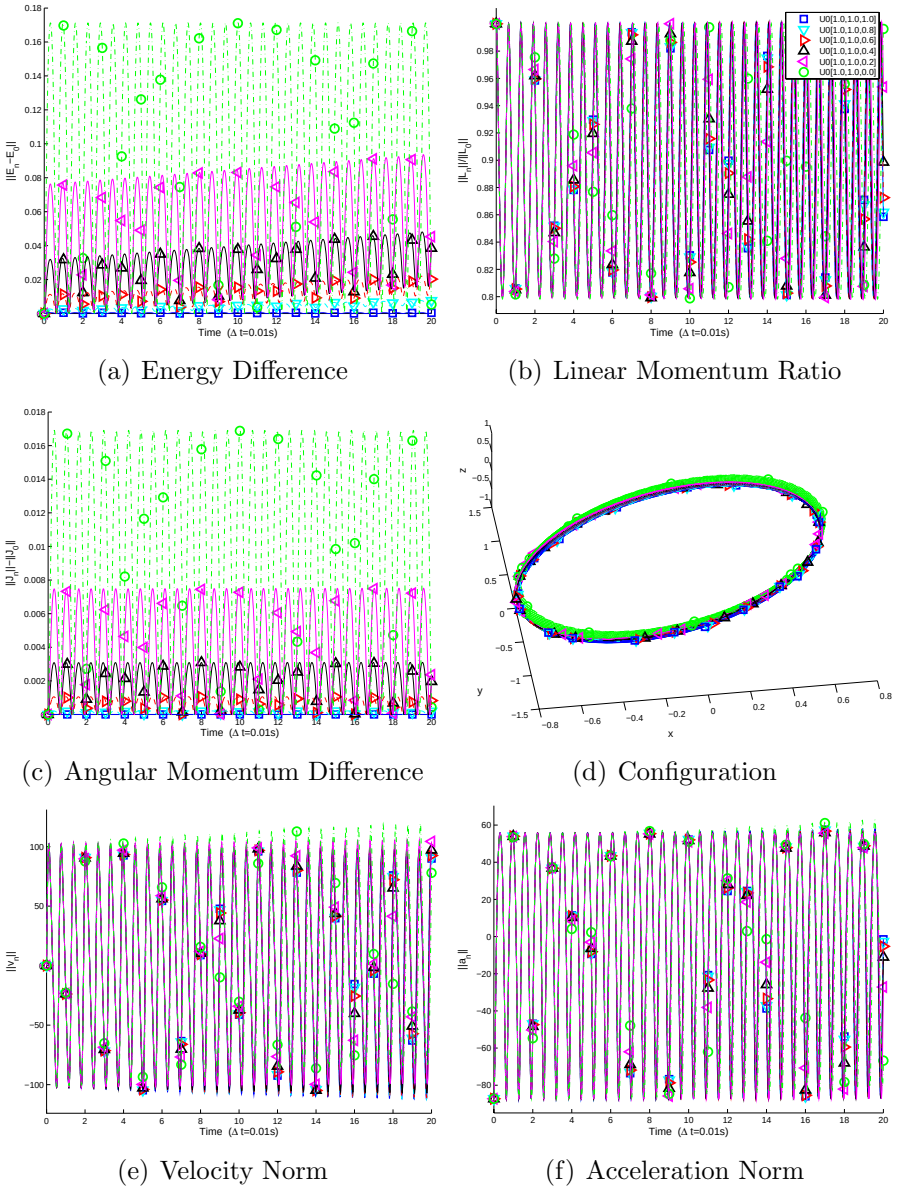


(a) Energy (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 7.17: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Implicit GSSSS family of algorithms (Option III) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]

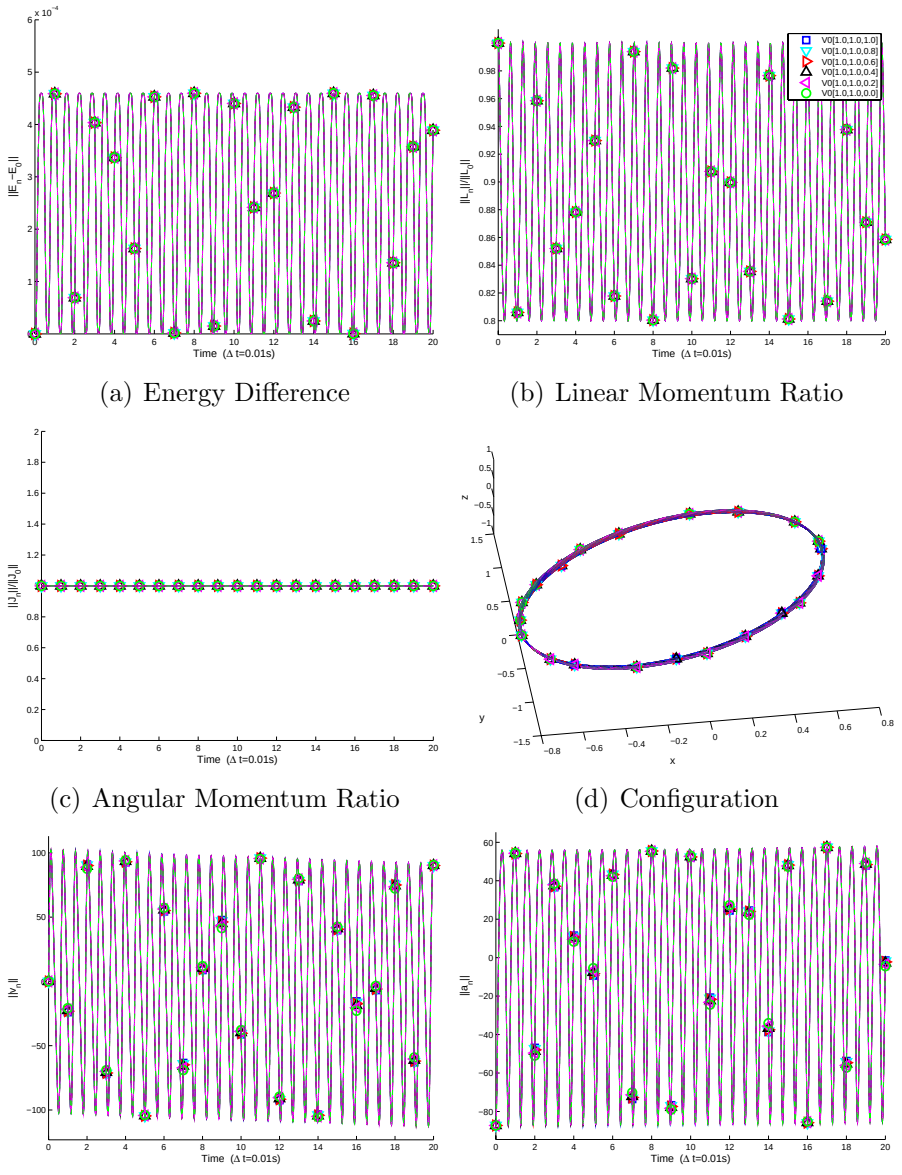


(a) Energy Difference (b) Linear Momentum Ratio

(c) Angular Momentum Difference (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 7.18: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Implicit GSSSS family of algorithms (Option III) - U0V0(1.0,1.0,ρ_∞)]

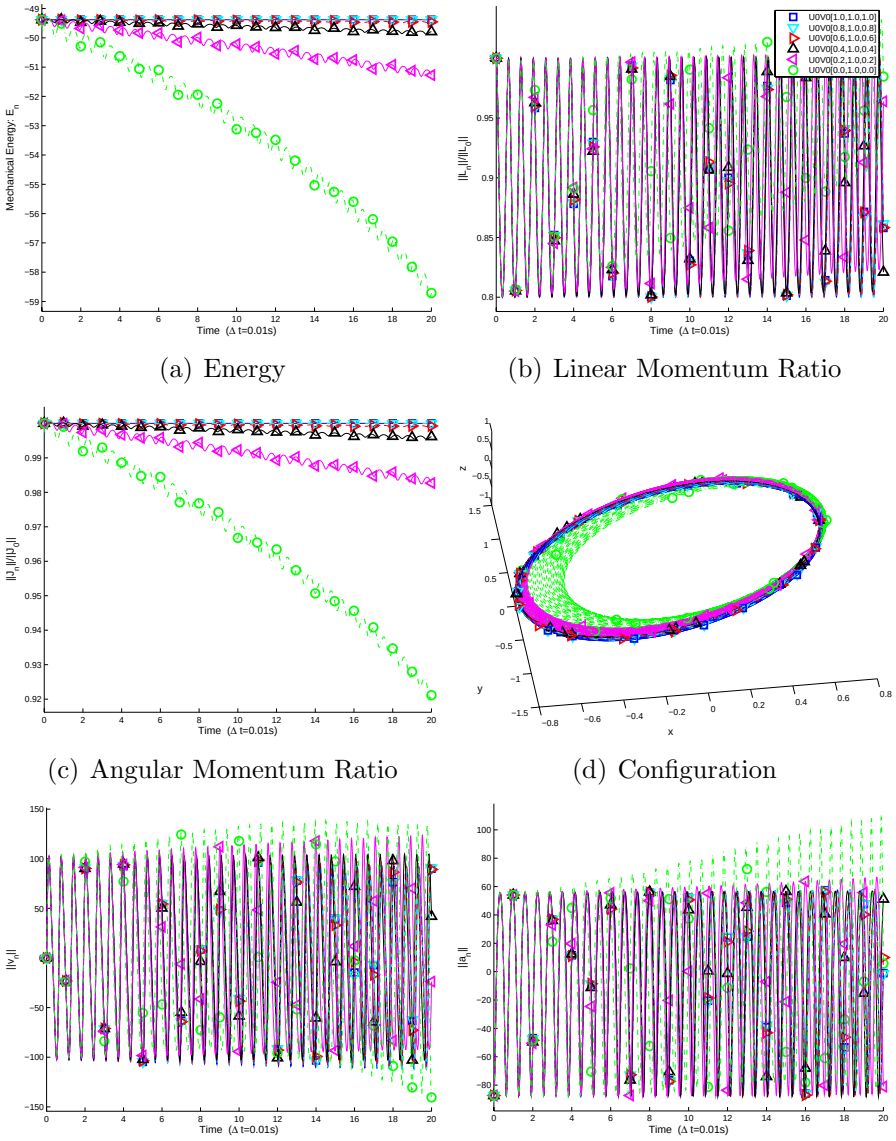


(a) Energy Difference (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 7.19: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Implicit GSSSS family of algorithms (Option III) - V0U0(1.0,1.0,ρ_∞)]

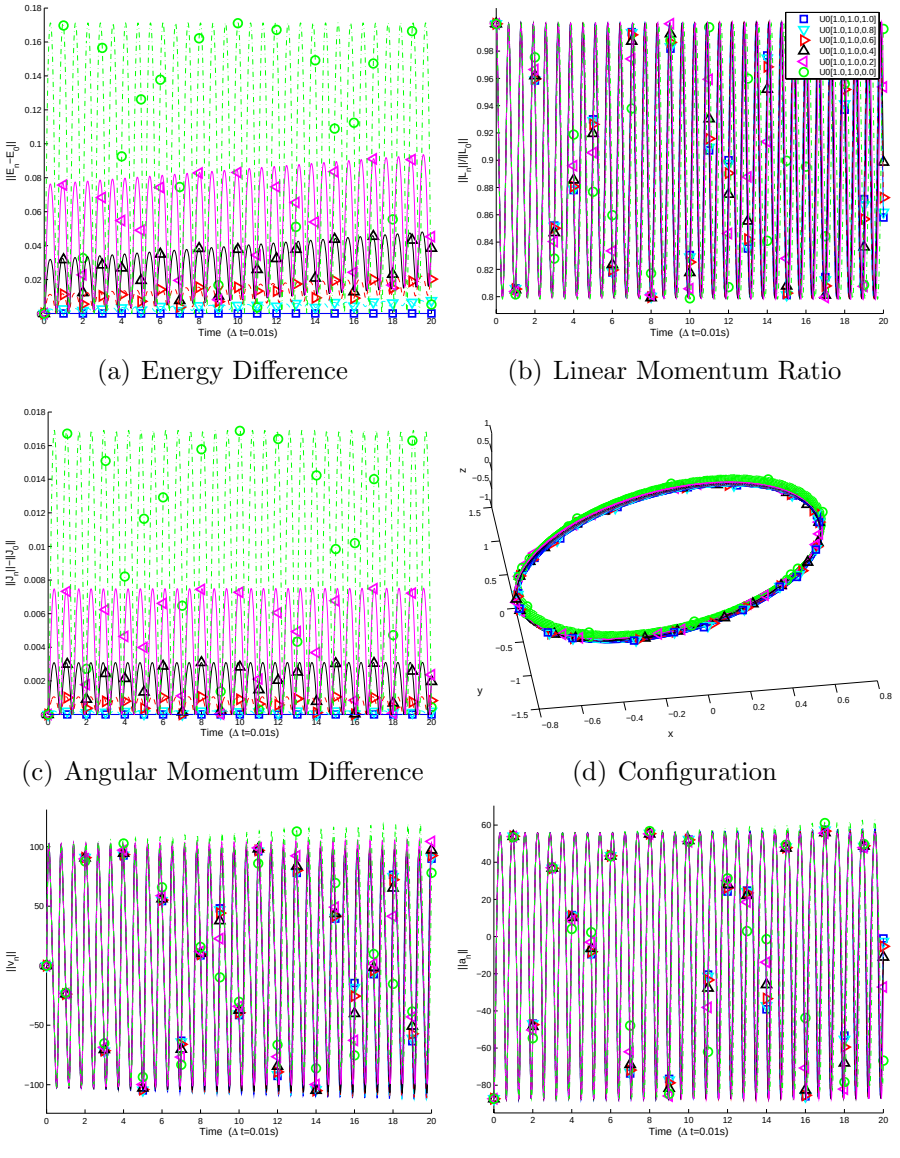


(a) Energy (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 7.20: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Implicit GSSSS family of algorithms (Modified Option III) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]

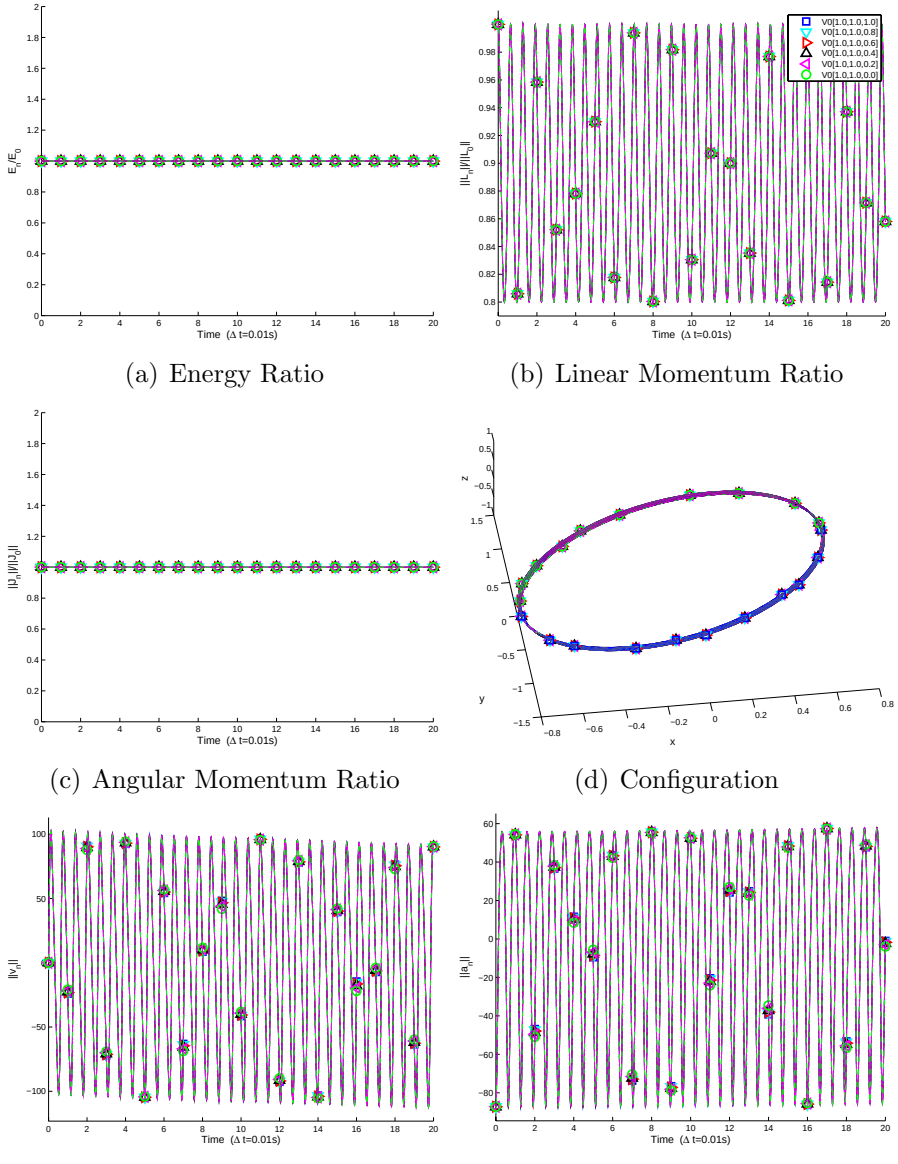


(a) Energy Difference (b) Linear Momentum Ratio

(c) Angular Momentum Difference (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 7.21: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Implicit GSSSS family of algorithms (Modified Option III) - U0V0(1.0,1.0,ρ_∞)]

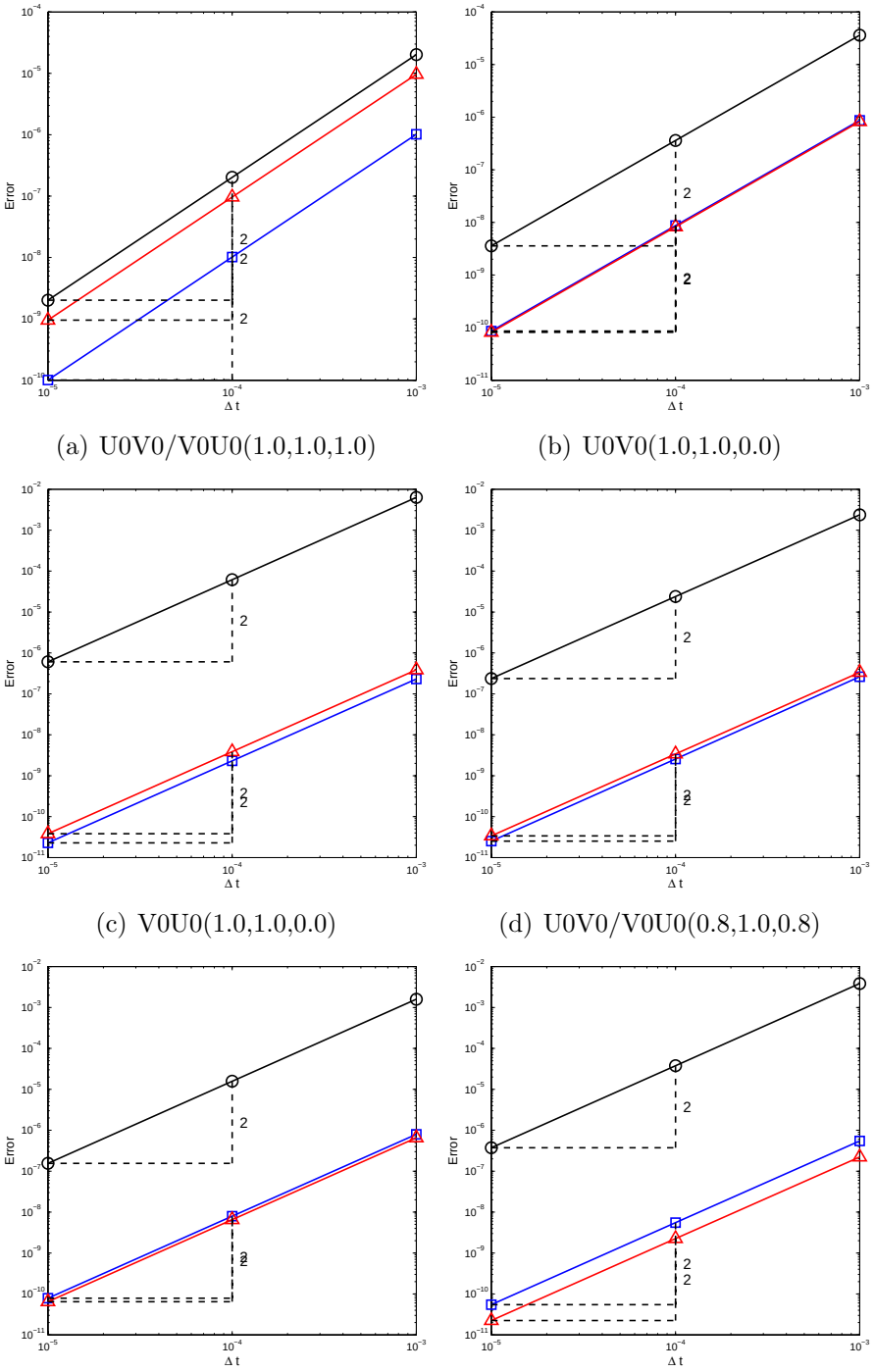


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 7.22: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Implicit GSSSS family of algorithms (Modified Option III) - V0U0(1.0,1.0,ρ_∞)]



(a) U0V0/V0U0(1.0,1.0,1.0)

(b) U0V0(1.0,1.0,0.0)

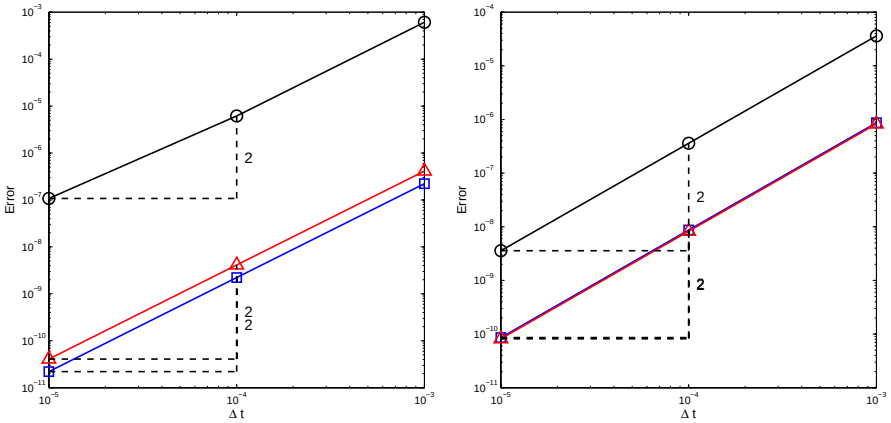
(c) V0U0(1.0,1.0,0.0)

(d) U0V0/V0U0(0.8,1.0,0.8)

(e) U0V1(0.6,0.9,0.2)

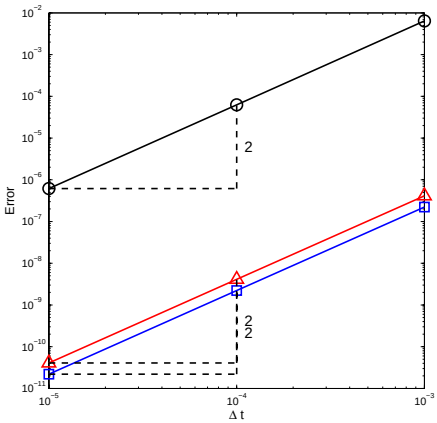
(f) V0U1(0.6,0.9,0.2)

Figure 7.23: Time accuracies in the configuration, velocity, and acceleration in the forced mechanical system. [Problem: Lennard-Jones potential problem] [Algorithm: Implicit GSSSS family of algorithms (Option III)]

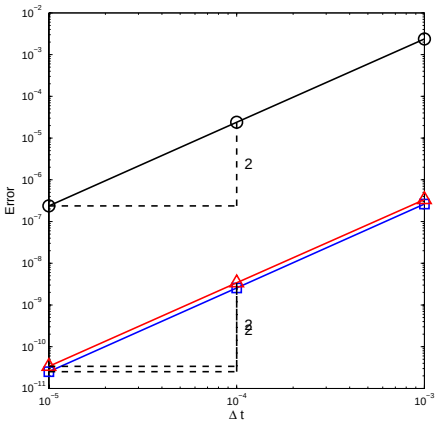


(a) U0V0/V0U0(1.0,1.0,1.0)

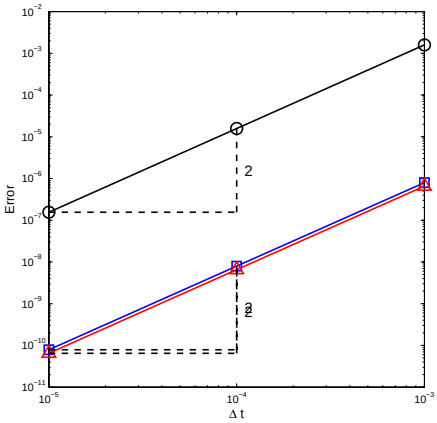
(b) U0V0(1.0,1.0,0.0)



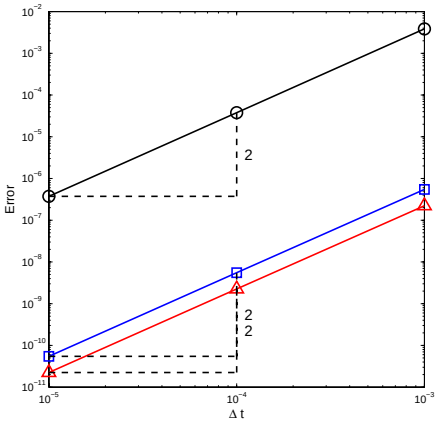
(c) V0U0(1.0,1.0,0.0)



(d) U0V0/V0U0(0.8,1.0,0.8)

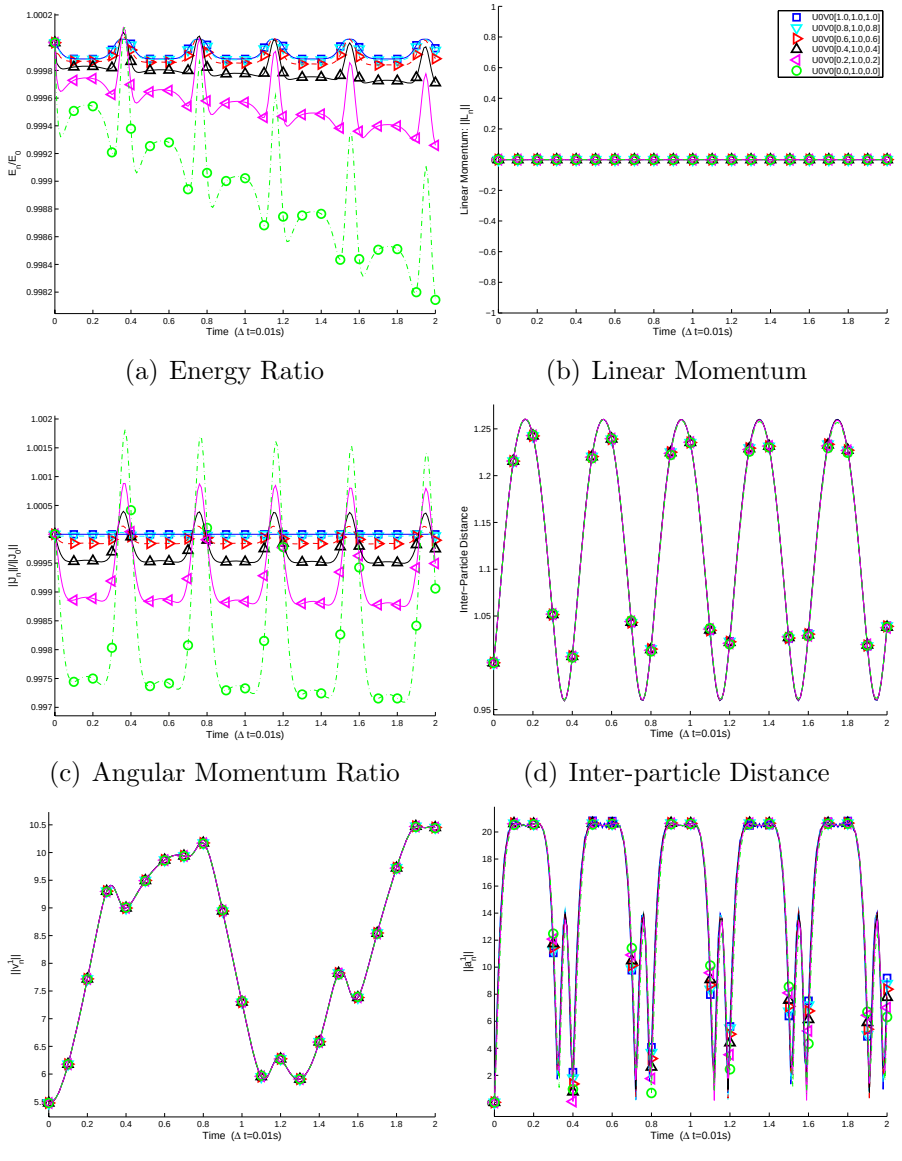


(e) U0V1(0.6,0.9,0.2)



(f) V0U1(0.6,0.9,0.2)

Figure 7.24: Time accuracies in the configuration, velocity, and acceleration in the forced mechanical system. [Problem: Lennard-Jones potential problem] [Algorithm: Implicit GSSSS family of algorithms (Modified Option III)]



(a) Energy Ratio

(b) Linear Momentum

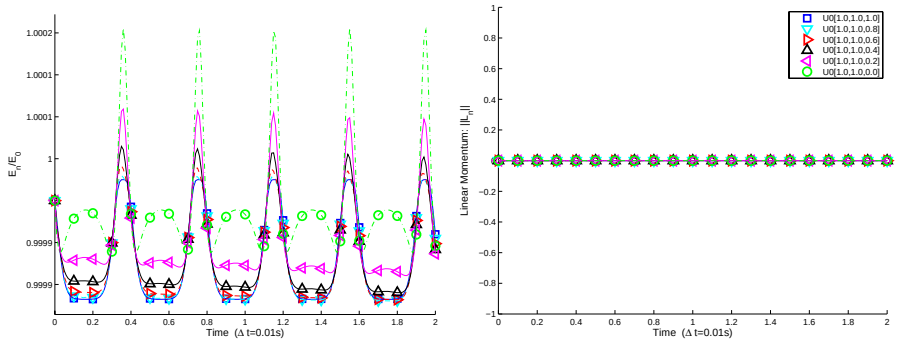
(c) Angular Momentum Ratio

(d) Inter-particle Distance

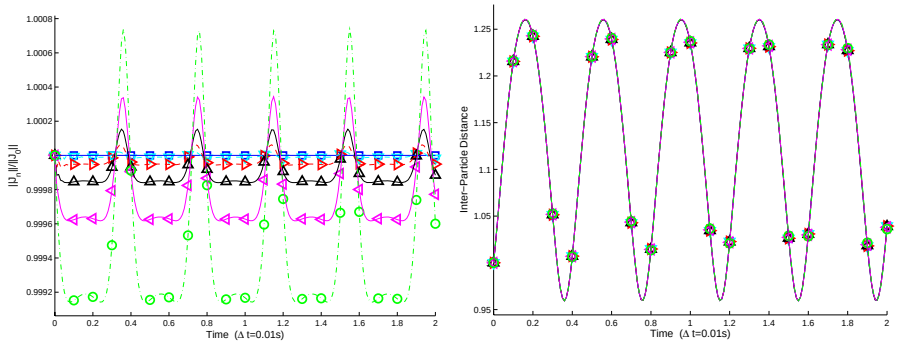
(e) Velocity Norm

(f) Acceleration Norm

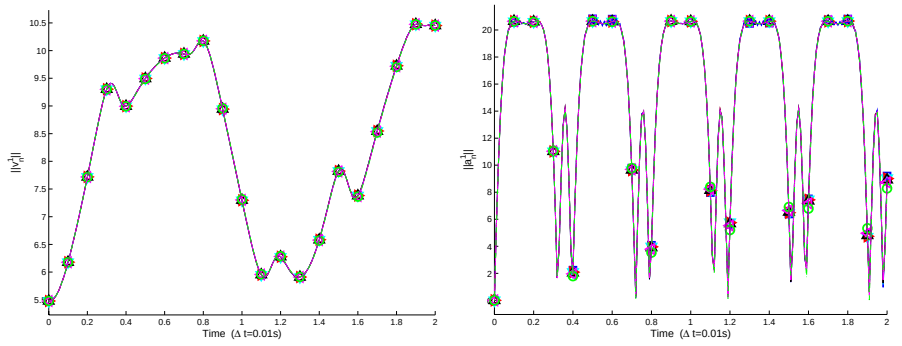
Figure 7.25: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Implicit GSSSS family of algorithms (Option III) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum

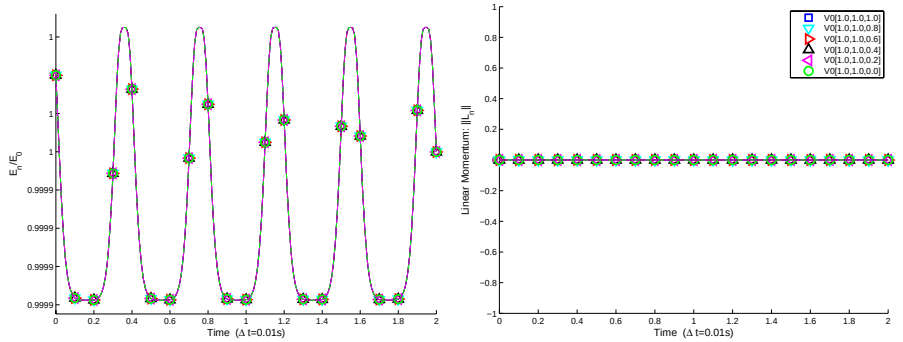


(c) Angular Momentum Ratio (d) Inter-particle Distance



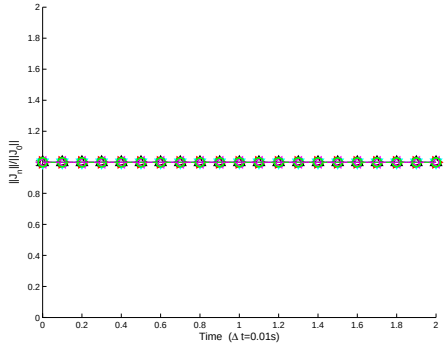
(e) Velocity Norm (f) Acceleration Norm

Figure 7.26: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Implicit GSSSS family of algorithms (Option III) - U0V0(1.0,1.0,ρ_∞)]

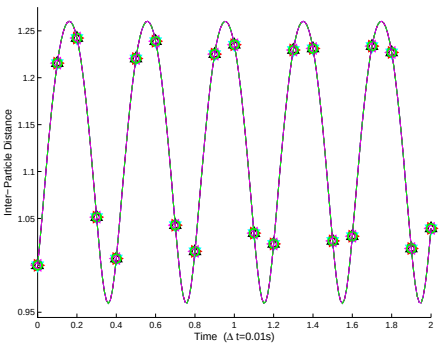


(a) Energy Ratio

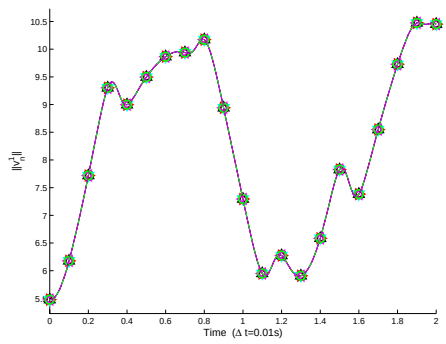
(b) Linear Momentum



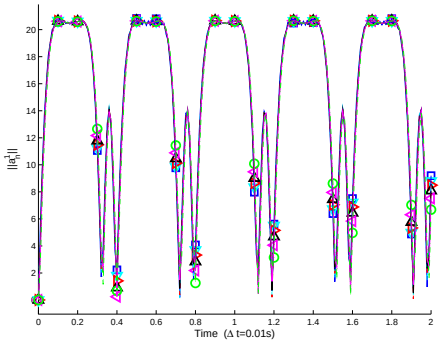
(c) Angular Momentum Ratio



(d) Inter-particle Distance

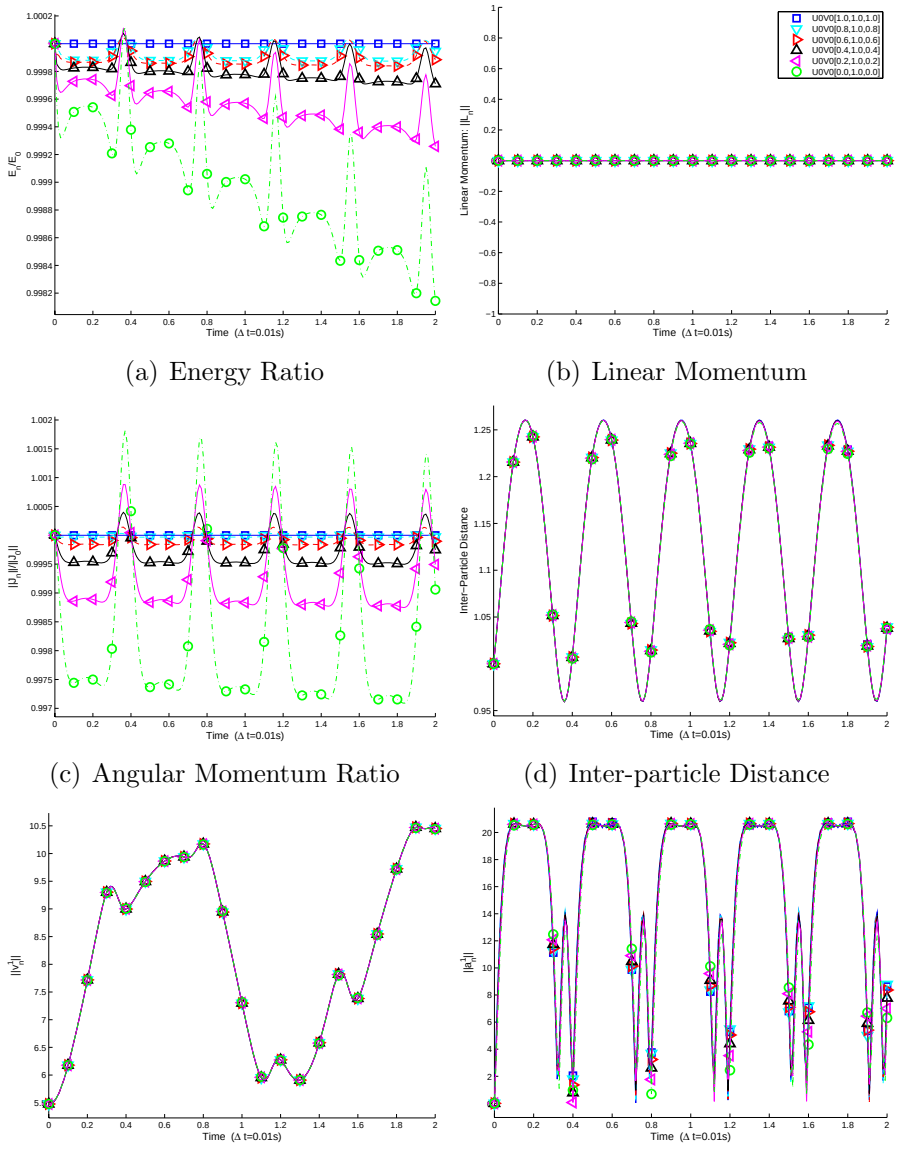


(e) Velocity Norm



(f) Acceleration Norm

Figure 7.27: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Implicit GSSSS family of algorithms (Option III) - V0U0(1.0,1.0,ρ_∞)]



(a) Energy Ratio

(b) Linear Momentum

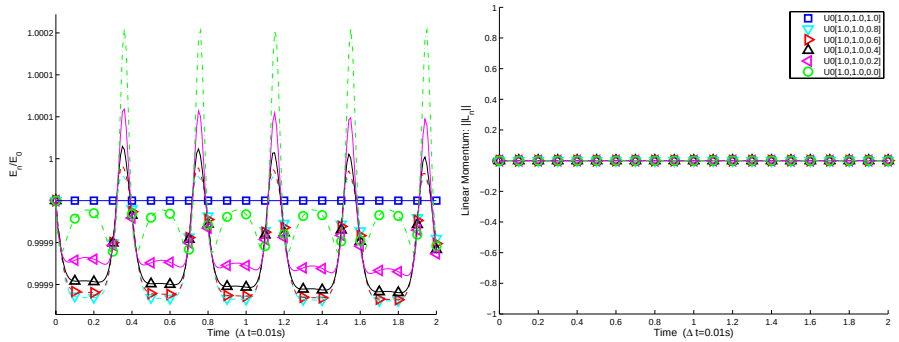
(c) Angular Momentum Ratio

(d) Inter-particle Distance

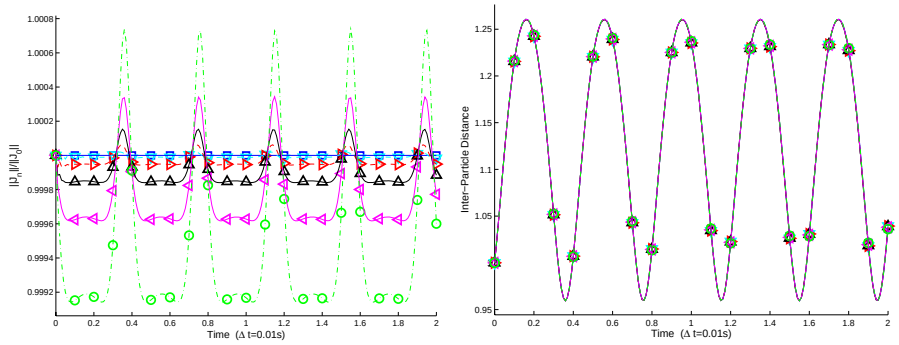
(e) Velocity Norm

(f) Acceleration Norm

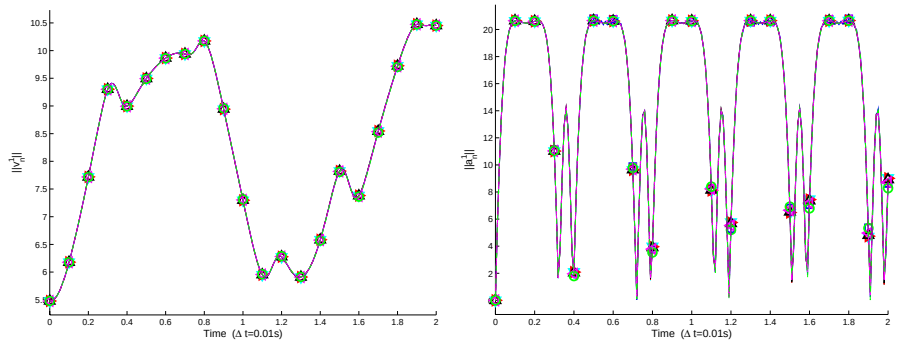
Figure 7.28: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Implicit GSSSS family of algorithms (Modified Option III) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum

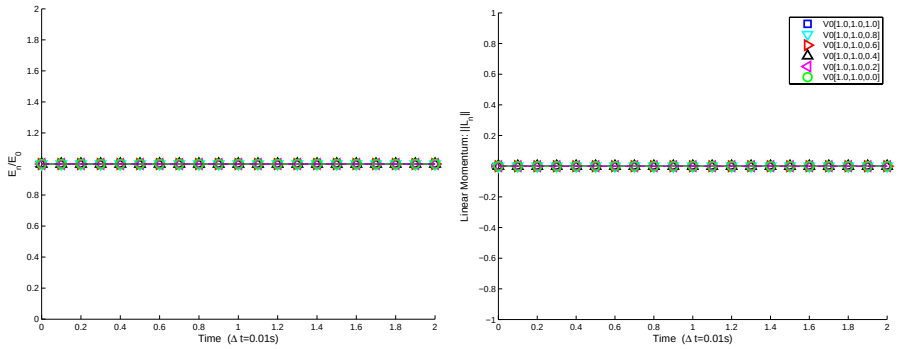


(c) Angular Momentum Ratio (d) Inter-particle Distance



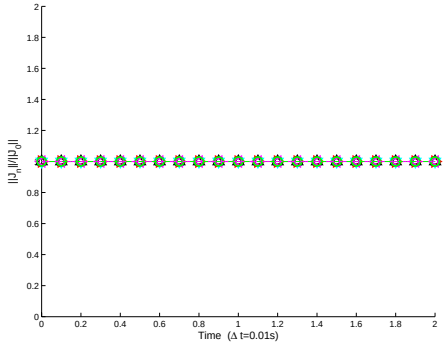
(e) Velocity Norm (f) Acceleration Norm

Figure 7.29: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Implicit GSSSS family of algorithms (Modified Option III) - U0V0(1.0,1.0,ρ_∞)]

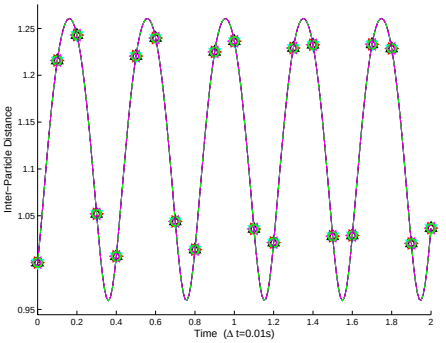


(a) Energy Ratio

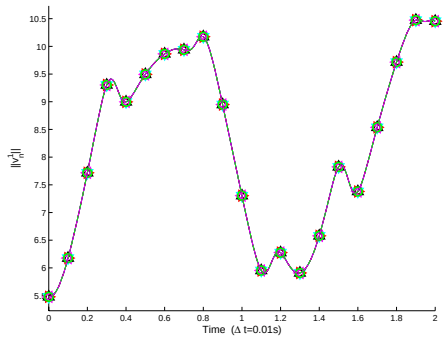
(b) Linear Momentum



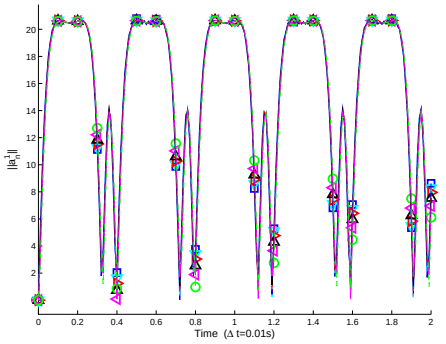
(c) Angular Momentum Ratio



(d) Inter-particle Distance



(e) Velocity Norm



(f) Acceleration Norm

Figure 7.30: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Implicit GSSSS family of algorithms (Modified Option III) - V0U0(1.0,1.0,ρ_∞)]

Chapter 8

Predictor-Corrector and General Explicit Generalized Single Step Single Solve Algorithms for Nonlinear Dynamical Systems

In this chapter, we will show: (i) how we can design and generate the explicit forms of the GSSSS family of algorithms from the implicit forms of the GSSSS family of algorithms in nonlinear dynamical systems, and (ii) how we can relax the unconditional stability limitations, and then define a more general form of representations which are in addition to those described in (i). Finally, we show a general algorithmic structure and framework where both architectures are unified under a single umbrella. The extension to the explicit schemes will be done based upon the theorem of the algorithmic time level consistency; hence, the resulting schemes will all maintain the same order of accuracy, namely, second-order time accuracy which is of fundamental interest to solving practical real-world problems. As can be seen from the PCE-GSSSS algorithms in the two-field form for linear dynamical systems, there exists no second-order time accurate algorithm for both cases of implicit and explicit treatment of physical damping. This scenario is still

true for nonlinear dynamical systems. Since we are primarily interested in the algorithm designs of second-order time accuracy, we will not show the nonlinear PCE-GSSSS algorithms in the two-field form, but focus attention only on the single-field form PCE-GSSSS family of algorithms as well as a more general representation which are in addition to those in the PCE-GSSSS framework. A sound theoretical basis is also provided to properly enable extensions of linear dynamic algorithms to nonlinear dynamic algorithms.

8.1 Nonlinear PCE-GSSSS and E-GSSSS Algorithms in Single-Field Form

In this section, we show three different frameworks of the explicit GSSSS family of algorithms, namely, Options I, II, and III. The classifications are based upon the discretization of the applied force fields, which includes the internal and external force fields; and they are similar to the concepts of Options I, II, and III for the designs of the implicit GSSSS families of algorithms. The details are shown below.

8.1.1 Algorithm Designs: Option I

From the implicit form of the GSSSS family of algorithms for linear dynamical systems in the sense of *Option I*, we may be able to construct two different forms of the explicit schemes. That is, the applied force, $\mathbf{f}^{\text{appl}}(\mathbf{q}, \dot{\mathbf{q}}, t) : Q \times T_{\mathbf{q}}Q \times \mathbb{I} \rightarrow \mathbb{R}^{n_{\text{dof}}}$, is discretized by either of the following ways, keeping the second-order time accuracy of the algorithms:

$$\tilde{\mathbf{f}}^{\text{appl}} = \mathbf{f}_n^{\text{appl}} + W_1 \Delta t \dot{\mathbf{f}}_n^{\text{appl}} + \frac{\Delta t^2 W_2}{2} \ddot{\mathbf{f}}_n^{\text{appl}} \tag{8.1}$$

or

$$\tilde{\mathbf{f}}^{\text{appl}} = \mathbf{f}_n^{\text{appl}} + W_3 (\mathbf{f}_{n+1}^{\text{appl}} - \mathbf{f}_n^{\text{appl}}) + (W_1 - W_3) \Delta t \dot{\mathbf{f}}_n^{\text{appl}} \tag{8.2}$$

where the applied force at time $t = t_{n+1}$ is defined by

$$\begin{aligned} \mathbf{f}_{n+1}^{\text{appl}} &= \mathbf{f}^{\text{appl}}(\dot{\mathbf{q}}_n + \lambda_4 \Delta t \ddot{\mathbf{q}}_n, \mathbf{q}_n + \lambda_1 \Delta t \dot{\mathbf{q}}_n + \lambda_2 \Delta t^2 \ddot{\mathbf{q}}_n, t_{n+1}) \\ &= \mathbf{f}^{\text{appl}}(\dot{\mathbf{q}}_n + \Delta t \ddot{\mathbf{q}}_n, \mathbf{q}_n + \Delta t \dot{\mathbf{q}}_n + \frac{\Delta t^2}{2} \ddot{\mathbf{q}}_n, t_{n+1}) \end{aligned} \tag{8.3}$$

Notice that both cases satisfy $\|\tilde{\mathbf{f}}^{\text{appl}} - \mathbf{f}^{\text{appl}}(t_{n+W_1})\| = \mathcal{O}(\Delta t^2)$. Although both cases lead to second-order accurate explicit schemes, it is not preferable to have the second total time derivative of the applied force due to its difficulty with numerical computations. Hence, we focus upon the second case. Consequently, the time integrator becomes

$$\mathbf{M}\tilde{\mathbf{a}} = \tilde{\mathbf{f}}^{\text{appl}} = \mathbf{f}_n^{\text{appl}} + W_3(\mathbf{f}_{n+1}^{\text{appl}} - \mathbf{f}_n^{\text{appl}}) + (W_1 - W_3)\Delta t \dot{\mathbf{f}}_n^{\text{appl}} \tag{8.4}$$

Hence, the acceleration increment is obtained as

$$\Delta \mathbf{a} = \frac{1}{W_1 \Lambda_6} \left[\mathbf{M}^{-1} \tilde{\mathbf{f}}^{\text{appl}} - \ddot{\mathbf{q}}_n \right] \tag{8.5}$$

In order to obtain the configuration, velocity, and acceleration at the next time step, keeping the second-order of time accuracy of the algorithms, we use the following updates:

$$\begin{aligned} \mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_1 \dot{\mathbf{q}}_n \Delta t + \lambda_2 \ddot{\mathbf{q}}_n \Delta t^2 + \lambda_3 \Delta \mathbf{a} \Delta t^2 \\ \dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4 \ddot{\mathbf{q}}_n \Delta t + \lambda_5 \Delta \mathbf{a} \Delta t \\ \ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_n + \Delta \mathbf{a} \end{aligned} \tag{8.6}$$

Alternatively, we can employ

$$\begin{aligned} \mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_1 \dot{\mathbf{q}}_n \Delta t + \lambda_2 \ddot{\mathbf{q}}_n \Delta t^2 \\ \dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4 \ddot{\mathbf{q}}_n \Delta t + \lambda_5 \Delta \mathbf{a} \Delta t \\ \ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_n + \Delta \mathbf{a} \end{aligned} \tag{8.7}$$

The difference between the above two updates is the appearance of $\lambda_3 \Delta \mathbf{a} \Delta t^2$ term in the update for the configuration. Note that we obtain only first-order time accuracy if we eliminate $\lambda_5 \Delta \mathbf{a} \Delta t$ term for the velocity updates. We, of course,

will get different numerical results by using the different updates shown above. The key idea is that since the general structures of the algorithm is similar, we combine both PCE- and E-GSSSS family of algorithms into a unified architecture although they are designed differently and separately; the only differences are in the [DNA] algorithmic parameters where, in the former, the [DNA] is in terms of $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}})$, and, in the latter, the [DNA] is in terms of (ρ_b, ρ_{3b}), where ρ_b and ρ_{3b} denote the principal root and the spurious root at the bifurcation point, respectively. Note that since $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}})$ are no longer the principal roots and spurious root at infinity for the explicit schemes, and they do not need to satisfy the inequality,

$$0 \leq \rho_\infty^{\mathrm{s}} \leq \rho_\infty^{\min} \leq \rho_\infty^{\max} \leq 1 \tag{8.8}$$

That is, they are to be treated as just the algorithmic parameters, ρ_∞^{s}, $\rho_\infty^{\min}$, $\rho_\infty^{\max} \in \mathbb{R}_+$.

Algorithm 28

Single-field Form PCE- and E-GSSSS Family of Algorithms for Nonlinear Dynamical Systems: Option I

Integrator:

$$\mathbf{M}\tilde{\mathbf{a}} = \tilde{\mathbf{f}}^{\mathrm{appl}}$$

where

$$\begin{aligned}
\tilde{\mathbf{f}}^{\mathrm{appl}} &= \mathbf{f}_n^{\mathrm{appl}} + W_3(\mathbf{f}_{n+1}^{\mathrm{appl}} - \mathbf{f}_n^{\mathrm{appl}}) + (W_1 - W_3)\Delta t \dot{\mathbf{f}}_n^{\mathrm{appl}} \\
\tilde{\mathbf{v}} &= \dot{\mathbf{q}}_n + W_1\Lambda_4\ddot{\mathbf{q}}_n\Delta t + W_2\Lambda_5\eta_1\Delta\mathbf{a}\Delta t \\
\tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1\Lambda_6\Delta\mathbf{a}
\end{aligned} \tag{8.9}$$

Updates:

$$\begin{aligned}
\mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + \lambda_3\eta_3\Delta\mathbf{a}\Delta t^2 \\
\dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5\Delta\mathbf{a}\Delta t \\
\ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_n + \Delta\mathbf{a}
\end{aligned} \tag{8.10}$$

Initial conditions:

$$\mathbf{q}(t_0) = \mathbf{q}_0$$
$$\dot{\mathbf{q}}(t_0) = \dot{\mathbf{q}}_0$$

Algorithmic parameters (for second-order time accuracy):
For U0 family-based PCE-GSSSS Algorithms:

$$W_1\Lambda_1 = \frac{1}{1+\rho_\infty^{\mathrm{s}}} \quad , \quad \lambda_1 = 1$$
$$W_2\Lambda_2 = \frac{1}{2(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_2 = \frac{1}{2}$$
$$W_3\Lambda_3 = \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_3 = \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})}$$
$$W_1\Lambda_4 = \frac{1}{1+\rho_\infty^{\mathrm{s}}} \quad , \quad \lambda_4 = 1$$
$$W_2\Lambda_5 = \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_5 = \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})}$$
$$W_1\Lambda_6 = \frac{2+\rho_\infty^{\min}+\rho_\infty^{\max}+\rho_\infty^{\mathrm{s}}-\rho_\infty^{\min}\rho_\infty^{\max}\rho_\infty^{\mathrm{s}}}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})}$$

For V0 family-based PCE-GSSSS Algorithms:

$$W_1\Lambda_1 = \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \quad , \quad \lambda_1 = 1$$
$$W_2\Lambda_2 = \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \quad , \quad \lambda_2 = \frac{1}{2}$$
$$W_3\Lambda_3 = \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_3 = \frac{1}{2(1+\rho_\infty^{\mathrm{s}})}$$
$$W_1\Lambda_4 = \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \quad , \quad \lambda_4 = 1$$
$$W_2\Lambda_5 = \frac{2}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_5 = \frac{1}{1+\rho_\infty^{\mathrm{s}}}$$
$$W_1\Lambda_6 = \frac{2+\rho_\infty^{\min}+\rho_\infty^{\max}+\rho_\infty^{\mathrm{s}}-\rho_\infty^{\min}\rho_\infty^{\max}\rho_\infty^{\mathrm{s}}}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})}$$

For E-GSSSS Algorithms:

$$
\begin{aligned}
W_1\Lambda_6 &= \frac{2+\rho_{3b}-\rho_{3b}\rho_b}{(1+\rho_b)(1+\rho_{3b})} \\
W_1\Lambda_1+\lambda_5 &= \frac{5+3\rho_{3b}+\rho_b-\rho_{3b}\rho_b}{2(1+\rho_b)(1+\rho_{3b})} \\
W_2\Lambda_2+\lambda_3 &= \frac{\rho_{3b}\rho_b-\rho_{3b}-2}{2\Lambda_6^2(1+\rho_{3b})^2(1+\rho_b)^2}\,[\,2(\rho_{3b}\rho_b-\rho_{3b}-2) \\
&\quad -\Lambda_6\,(\rho_{3b}\rho_b-3\rho_{3b}-\rho_b-5)\,] \\
&\quad -\frac{5+3\rho_b+\rho_{3b}(1-\rho_b)(4+\rho_{3b}+2\rho_b)}{(\rho_{3b}\rho_b-\rho_{3b}-2)(1+\rho_{3b})(1+\rho_b)^2} \\
\lambda_1 &= \lambda_4 = 1,\ \lambda_2 = \frac{1}{2},\ \Lambda_1 = \Lambda_4
\end{aligned}
$$

Remark 28 (Algorithm 28)

1. ***Time Level Consistency:*** *Assuming $\lambda_i = \Lambda_i$ (for $i = 1, 2, \cdots, 5$), the algorithmic time level of the discrete equation of motion in Algorithm 28 is $t^* = t_{n+W_1}$. Note that schemes in Algorithm 28 have second-order time accuracies in $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\ddot{\mathbf{q}}$ regardless of the choice of the spectral conditions.*

2. ***Conservation of Linear Momentum:*** *The discrete total linear momentum within the time step is exactly conserved in the absence of the external loads for any choices of the algorithmic parameters, (η_1, η_3), $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}})$, and (ρ_b, ρ_{3b}).*

3. ***Conservation of Angular Momentum:*** *When we select $\eta_3 = 0$ in the configuration update, the total angular momentum within the time step $[t_n, t_{n+1}]$ is exactly conserved in the absence of the external loads for any spectral conditions of $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\mathrm{s}} \in [0, 1]$ within the U0 family-based PCE-GSSSS algorithms. The discrete total linear momentum within the time step is exactly conserved in the absence of the external loads for any spectral conditions. No member in the V0 family satisfies the conservation of the discrete angular momentum within the time step.*

4. *The algorithmic applied force* $\tilde{\mathbf{f}}^{\text{appl}}$ *is reduced to*

$$\tilde{\mathbf{f}}^{\text{appl}} = \mathbf{f}_n^{\text{appl}} + W_1(\mathbf{f}_{n+1}^{\text{appl}} - \mathbf{f}_n^{\text{appl}}) =: \mathbf{f}_{n+W_1}^{\text{appl}} \tag{8.11}$$

for any spectral conditions, with $\lambda_i = \Lambda_i$ *(for* $i = 1, 2, \cdots, 5$*), within the U0 family-based PCE-GSSSS algorithms, but, we must have* $\rho_\infty^{\max} = 1$ *within the V0 family-based PCE-GSSSS algorithms.*

8.1.2 Algorithm Designs: Option II

Extension of the implicit GSSSS family of algorithms to nonlinear dynamical systems in the sense of *Option II* to the explicit forms is fairly straightforward. In the evaluation of the applied force, we take the following

$$\begin{aligned}\tilde{\mathbf{f}}^{\text{appl}} &= \mathbf{f}^{\text{appl}}(\mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2, \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t, t_{n+W_1}) \\ &= \mathbf{f}^{\text{appl}}(\mathbf{q}_n + \dot{\mathbf{q}}_n\Delta t + \frac{\Delta t^2}{2}\ddot{\mathbf{q}}_n, \dot{\mathbf{q}}_n + \ddot{\mathbf{q}}_n\Delta t, t_{n+W_1})\end{aligned} \tag{8.12}$$

for the implicit treatment of the velocity term and the following

$$\begin{aligned}\tilde{\mathbf{f}}^{\text{appl}} &= \mathbf{f}^{\text{appl}}(\mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2, \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5\Delta\mathbf{a}\Delta t, t_{n+W_1}) \\ &= \mathbf{f}^{\text{appl}}(\mathbf{q}_n + \dot{\mathbf{q}}_n\Delta t + \frac{\Delta t^2}{2}\ddot{\mathbf{q}}_n, \dot{\mathbf{q}}_n + \ddot{\mathbf{q}}_n\Delta t + \lambda_5\Delta\mathbf{a}\Delta t, t_{n+W_1})\end{aligned} \tag{8.13}$$

for the explicit treatment of the velocity term. The choice of the corresponding updates is either Eq. (8.6) or Eq. (8.7).

Algorithm 29

Single-field Form PCE- and E-GSSSS Family of Algorithms for Nonlinear Dynamical Systems: Option II

Integrator:

$$\mathbf{M}\tilde{\mathbf{a}} = \mathbf{f}^{\text{appl}}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}, t_{n+W_1})$$

where

$$\begin{aligned}\tilde{\mathbf{q}} &= \mathbf{q}_n + W_1\Lambda_1\dot{\mathbf{q}}_n\Delta t + W_2\Lambda_2\ddot{\mathbf{q}}_n\Delta t^2 \\ \tilde{\mathbf{v}} &= \dot{\mathbf{q}}_n + W_1\Lambda_4\ddot{\mathbf{q}}_n\Delta t + W_2\Lambda_5\eta_1\Delta\mathbf{a}\Delta t \\ \tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1\Lambda_6\Delta\mathbf{a}\end{aligned} \tag{8.14}$$

Updates:

$$\begin{aligned}
\mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_1 \dot{\mathbf{q}}_n \Delta t + \lambda_2 \ddot{\mathbf{q}}_n \Delta t^2 + \lambda_3 \eta_3 \Delta \mathbf{a} \Delta t^2 \\
\dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4 \ddot{\mathbf{q}}_n \Delta t + \lambda_5 \Delta \mathbf{a} \Delta t \\
\ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_n + \Delta \mathbf{a}
\end{aligned} \tag{8.15}$$

Initial conditions:

$$\begin{aligned}
\mathbf{q}(t_0) &= \mathbf{q}_0 \\
\dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0
\end{aligned}$$

Algorithmic parameters (for second-order time accuracy):
Employ from Algorithm 28

Remark 29 (Algorithm 29)

1. ***Time Level Consistency:*** *Assuming $\lambda_i = \Lambda_i$ (for $i = 1, 2, \cdots, 5$), the algorithmic time level of the discrete equation of motion in Algorithms 29 is $t^* = t_{n+W_1}$. Again, all schemes in Algorithms 29 have second-order time accuracies in $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\ddot{\mathbf{q}}$ regardless of the choice of the spectral conditions.*

2. ***Conservation of Linear Momentum:*** *The discrete total linear momentum within the time step is exactly conserved in the absence of the external loads for any choices of the algorithmic parameters, (η_1, η_3), $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}})$, and (ρ_b, ρ_{3b}).*

3. ***Conservation of Angular Momentum:*** *When we select $\eta_3 = 0$ in the configuration update, the total angular momentum within the time step $[t_n, t_{n+1}]$ is exactly conserved in the absence of the external loads only for the central difference method, i.e., $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\mathrm{s}} = 0$ in the U0 family-based PCE-GSSSS algorithms or $\rho_{3b} = \rho_b = 1$ in the general explicit family of algorithms. None of the algorithms belong to the V0 family-based PCE-GSSSS algorithms conserves the discrete angular momentum exactly. On the otherhand, the discrete total linear momentum within the time step is exactly conserved in the absence of the external loads for any spectral conditions.*

4. *By selecting* $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}}) = (1, 1, \rho_\infty^{\mathrm{s}})$ *with* $\rho_\infty^{\mathrm{s}} \in [0, 1]$ *in the U0 family-based PCE-GSSSS algorithms* ($\lambda_i = \Lambda_i$)*, we obtain* $W_1 = 1/(1 + \rho_\infty^{\mathrm{s}}) \in [1/2, 1]$ *and*

$$\mathbf{M}\tilde{\mathbf{a}} = \mathbf{f}^{\mathrm{appl}}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}, t_{n+W_1})$$

where

$$\begin{aligned} \tilde{\mathbf{q}} &= \mathbf{q}_n + W_1 \dot{\mathbf{q}}_n \Delta t + \frac{\Delta t^2}{2} W_1 \ddot{\mathbf{q}}_n \\ \tilde{\mathbf{v}} &= \dot{\mathbf{q}}_n + W_1 \ddot{\mathbf{q}}_n \Delta t + \frac{\Delta t}{2} W_1 \eta_1 \Delta \mathbf{a} \\ \tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1 \Delta \mathbf{a} =: \ddot{\mathbf{a}}_{n+W_1} \end{aligned} \tag{8.16}$$

Updates:

$$\begin{aligned} \mathbf{q}_{n+1} &= \mathbf{q}_n + \dot{\mathbf{q}}_n \Delta t + \frac{\Delta t^2}{2} \ddot{\mathbf{q}}_n + \frac{\Delta t^2}{4} \eta_3 \Delta \mathbf{a} \\ \dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \ddot{\mathbf{q}}_n \Delta t + \frac{\Delta t}{2} \Delta \mathbf{a} \\ \ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_n + \Delta \mathbf{a} \end{aligned} \tag{8.17}$$

The central difference method can be recovered by selecting $\eta_3 = 1$ *and* $\rho_\infty^{\mathrm{s}} = 1$*. Therefore, the algorithmic time level of the central difference method is* $t^* = t_{n+1}$*.*

5. ***Application to the Nonlinear Elastodynamical Systems:*** *Application of Algorithm 29 for the initial-value problem in the sense of the single-field form in the nonlinear elastodynamical system yields*

$$\begin{aligned} &\sum_{j=1}^{n_{\mathrm{node}}} M_{ij} \tilde{\mathbf{a}} + \int_{\mathcal{B}} \tilde{\mathbf{F}}^h \cdot \bar{\mathbf{S}}^h(\tilde{\mathbf{F}}^h) \ \mathrm{GRAD}\ N_i \, dV \\ &= \mathbf{F}_i^{\mathrm{diss}^h}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}) + \int_{\mathcal{B}} N_i \rho_0 \mathbf{B}(t_{n+W_1}) dV + \int_{\partial \mathcal{B}_\sigma} N_i \bar{\mathbf{T}}(t_{n+W_1}) dA \end{aligned} \tag{8.18}$$

with the updates and the initial conditions, where $\mathbf{F}^{\mathrm{diss}^h}$ *denotes the discrete dissipative force vector, and the algorithmic discrete deformation gradient*

tensor field $\tilde{\mathbf{F}}_n^h$ *is defined by*

$$\tilde{\mathbf{F}}^h := \mathbf{I} + \sum_{i=1}^{n_{\text{node}}} \tilde{\mathbf{q}}^i \otimes \text{GRAD}\, N_i(\mathbf{X}) \tag{8.19}$$

8.1.3 Algorithm Designs: Option III

Similar to the case of *Options I and II*, we can construct the explicit forms of *Option III* as follows. If the balance for the system of N particles is given by

$$m_i \mathbf{a}^i - \sum_{j \neq i}^{N} \mathscr{V}'_{ij}(r^{ij}) \frac{\mathbf{q}^{ij}}{r^{ij}} = \mathbf{f}_i^{\text{ext}}(\mathbf{q}, \dot{\mathbf{q}}, t) \tag{8.20}$$

where $r^{ij} = \|\mathbf{q}^{ij}\|$ denotes the inter-particle distance between particles i and j, as discussed before, we suggest

$$m_i \tilde{\mathbf{a}}^i - \sum_{j \neq i}^{N} \mathscr{V}'_{ij}(\tilde{r}^{ij}) \frac{\tilde{\mathbf{q}}^{ij}}{\tilde{r}^{ij}} = \mathbf{f}_i^{\text{ext}}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}, t_{n+W_1}) \tag{8.21}$$

where

$$\begin{aligned}
\tilde{r}^{ij} &= r_n^{ij} + W_3(r_{n+1}^{ij} - r_n^{ij}) + (W_1 - W_3)\Delta t \dot{r}_n^{ij} && (8.22)\\
\tilde{\mathbf{q}}^{ij} &= \mathbf{q}_n^{ij} + W_1\Lambda_1 \Delta t \dot{\mathbf{q}}_n^{ij} + W_2\Lambda_2 \Delta t^2 \ddot{\mathbf{q}}_n^{ij} && (8.23)\\
\tilde{\mathbf{v}} &= \dot{\mathbf{q}}_n + \Delta t W_1 \Lambda_4 \ddot{\mathbf{q}}_n && (8.24)\\
\tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1 \Lambda_6 \Delta \mathbf{a} && (8.25)
\end{aligned}$$

in which r_{n+1}^{ij} is evaluated as

$$\begin{aligned}
r_{n+1}^{ij} &= \left\| \mathbf{q}_n^{ij} + \lambda_1 \Delta t \dot{\mathbf{q}}_n^{ij} + \lambda_2 \Delta t^2 \ddot{\mathbf{q}}^{ij} \right\| \\
&= \| \mathbf{q}_n^{ij} + \Delta t \dot{\mathbf{q}}_n^{ij} + \frac{\Delta t^2}{2} \ddot{\mathbf{q}}^{ij} \|
\end{aligned} \tag{8.26}$$

Instead of Eq. (8.22), we can alternatively introduce the algorithmic inter-particle distance as

$$\tilde{r}^{ij} = r_n^{ij} + W_1 \Lambda_1 \Delta t \dot{r}_n^{ij} + W_2 \Lambda_2 \Delta t^2 \ddot{r}_n^{ij} \tag{8.27}$$

following the same manner for the configuration; however, calculating $\ddot{r}^{ij}$ at every time step is not efficient; therefore, Eq. (8.22) is preferred. But, it is important to note that both cases maintain the second-order accuracy of the algorithms due to $\tilde{r}^{ij} - r^{ij}(t_{n+W_1}) = \mathcal{O}(\Delta t^2)$. Once we obtain the acceleration increment, we can follow the updates shown in Eq. (8.6) or Eq. (8.7).

Algorithm 30

Single-field Form PCE- and E-GSSSS Family of Algorithms for Nonlinear Dynamical Systems: Option III

Integrator:

$$m_i\tilde{\mathbf{a}}^i - \sum_{j\neq i}^{N} \mathscr{V}'_{ij}(\tilde{r}^{ij})\frac{\tilde{\mathbf{q}}^{ij}}{\tilde{r}^{ij}} = \mathbf{f}_i^{\text{ext}}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}, t_{n+W_1})$$

where

$$\begin{aligned}
\tilde{r}^{ij} &= r_n^{ij} + W_3(r_{n+1}^{ij} - r_n^{ij}) + (W_1 - W_3)\Delta t\dot{r}_n^{ij} \\
\tilde{\mathbf{q}}^{ij} &= \mathbf{q}_n^{ij} + W_1\Lambda_1\Delta t\dot{\mathbf{q}}_n^{ij} + W_2\Lambda_2\Delta t^2\ddot{\mathbf{q}}_n^{ij} \\
\tilde{\mathbf{v}} &= \dot{\mathbf{q}}_n + \Delta t W_1\Lambda_4\ddot{\mathbf{q}}_n + W_2\Lambda_5\eta_1\Delta\mathbf{a}\Delta t \\
\tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1\Lambda_6\Delta\mathbf{a}
\end{aligned} \tag{8.28}$$

Updates:

$$\begin{aligned}
\mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + \lambda_3\eta_3\Delta\mathbf{a}\Delta t^2 \\
\dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5\Delta\mathbf{a}\Delta t \\
\ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_n + \Delta\mathbf{a}
\end{aligned} \tag{8.29}$$

Initial conditions:

$$\begin{aligned}
\mathbf{q}(t_0) &= \mathbf{q}_0 \\
\dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0
\end{aligned}$$

Algorithmic parameters (for second-order time accuracy):
Employ from Algorithm 28

Remark 30 (Algorithm 30)

1. ***Time Level Consistency:*** *Assuming $\lambda_i = \Lambda_i$ (for $i = 1, 2, \cdots, 5$), the algorithmic time level of the discrete equation of motion in Algorithm 30 is $t^* = t_{n+W_1}$. All schemes in Algorithm 30 have second-order time accuracies in $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\ddot{\mathbf{q}}$ regardless of the choice of the spectral conditions.*

2. ***Conservations of Linear Momentum:*** *The discrete total linear momentum within the time step is exactly conserved in the absence of the external loads for any choices of the algorithmic parameters, (η_1, η_3), $(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}})$, and (ρ_b, ρ_{3b}).*

3. ***Conservations of Angular Momentum:*** *When we select $\eta_3 = 0$ in the configuration update, the total angular momentum within the time step $[t_n, t_{n+1}]$ is exactly conserved in the absence of the external loads only for the central difference method in the sense of Option III, i.e., $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\mathrm{s}} = 0$ in the U0 family-based PCE-GSSSS algorithms or $\rho_{3b} = \rho_b = 1$ in the general explicit family of algorithms. None of the algorithms which belong to the V0 family-based PCE-GSSSS algorithms conserves the discrete angular momentum exactly.*

4. ***Application to the Nonlinear Elastodynamical Systems:*** *Applying the similar concept to the initial-value problem for the nonlinear elastodynamical system, the time integrator of the explicit version of the implicit family of GSSSS algorithms, including both the PCE- and E-GSSSS family of algorithms, in the sense of Option III yield the follows:*

$$\begin{aligned}&\sum_{j=1}^{n_{\mathrm{node}}} M_{ij}\tilde{\mathbf{a}} + \int_{\mathcal{B}} \tilde{\mathbf{F}}^h \cdot \hat{\mathbf{S}}^h(\tilde{\mathbf{C}}^h)\ \mathrm{GRAD}\ N_i\, dV \\ &= \mathbf{F}_i^{\mathrm{diss}^h}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}) + \int_{\mathcal{B}} N_i \rho_0 \mathbf{B}(t_{n+W_1}) dV + \int_{\partial \mathcal{B}_\sigma} N_i \bar{\mathbf{T}}(t_{n+W_1}) dA\end{aligned} \tag{8.30}$$

where

$$\tilde{\mathbf{C}}^h = \mathbf{C}_n^h + W_1(\mathbf{C}_{n+1}^h - \mathbf{C}_n^h) + (W_1 - W_3)\dot{\mathbf{C}}_n \Delta t \tag{8.31}$$

with the updates and the initial conditions, where $\mathbf{F}^{\mathrm{diss}^h}$ *denotes the discrete dissipative force vector.*

8.2 Numerical Results

In this section, numerical results using the various explicit families of algorithms presented in this chapter are presented. We again consider the following three problems: (1) Nonlinear oscillator [1], (2) (Classical) Kepler's problem, and (3) Lennard-Jones (5, 3) Potential 2-body Problem. In addition to these problems, the accuracy plots for the spring-pendulum problem are shown to emphasize that the second-order time accuracy of time integration schemes are obtained both for the implicit and explicit treatments of the velocity terms. The information of the input parameters and such are the same for the simulations as in Section 6.3. The time step size used is $\Delta t = 0.01$ sec. For the brief summaries of the numerical examples, please see Appendix A. The selected algorithms are:
Explicit GSSSS family of algorithms (Option I): Algorithm 28
Explicit GSSSS family of algorithms (Option II): Algorithm 29
Explicit GSSSS family of algorithms (Option III): Algorithm 30
Note that we have the explicit treatment of the velocity term (ET) if we select $\eta_1 = 0$. We show the results for both cases where $\eta_3 = 1$ and $\eta_3 = 0$ for each problem. For each algorithm, we select,
U0V0/V0U0(ρ_∞,1.0,ρ_∞)
U0V0(1.0,1.0,ρ_∞^{s})
V0U0(1.0,1.0,ρ_∞^{s})
where

$$\begin{aligned} &\rho_\infty = \rho_\infty^{\min} = \rho_\infty^{\max} \in \{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\} \\ &\rho_\infty^{\mathrm{s}} \in \{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\} \end{aligned} \tag{8.32}$$

The legends of the figures are shown in Fig. 4.5.

Nonlinear oscillator problem: The explicit families of algorithms in the sense of Options I, II, and III, presented in this chapter, have been designed and developed based upon the theorem of the consistent algorithmic time level; that is, the algorithms should maintain the second-order accuracy in time since the algorithmic time level is consistent at time $t = t_{n+W_1}$. As expected, the second-order time accuracies in the configuration, velocity, and acceleration are observed in the forced-damped system, Eq. (A.31) with $C = 0.001$ and $\mathbf{f}^{\text{ext}} = \bar{\mathbf{f}}\sin(t)$ where $\bar{\mathbf{f}} = (0.1, -0.5)^T$, with $\eta_1 = 0$; see Fig. 8.1, Fig. 8.2, and Fig. 8.3 for Option I, II, and III, respectively, with $\eta_3 = 1$; and see Fig. 8.22, Fig. 8.23, and Fig. 8.24 for Option I, II, and III, respectively, with $\eta_3 = 0$. Therefore, it should be noted that the second-order time accuracies in the configuration, velocity, and acceleration can be obtained also for the case where the velocity term is treated implicitly ($\eta_1 = 1$). Again, the configuration, velocity, and acceleration have been used to calculate the reference errors,

$$\begin{aligned} \text{error}_q &= \left| \frac{q^1 - q^1_{\text{ref}}}{q^1_{\text{ref}}} \right| \\ \text{error}_v &= \left| \frac{\dot{q}^1 - \dot{q}^1_{\text{ref}}}{\dot{q}^1_{\text{ref}}} \right| \\ \text{error}_a &= \left| \frac{\ddot{q}^1 - \ddot{q}^1_{\text{ref}}}{\ddot{q}^1_{\text{ref}}} \right| \end{aligned} \tag{8.33}$$

respectively, at time $t = 1$ sec. The reference configurations, velocities, and accelerations have been obtained with a sufficiently small time step size $\Delta t = 10^{-5}$.

In this oscillator problem, we observe the time responses of the mechanical energy and linear and angular momenta; and configuration, velocity and acceleration norms, i.e., $\|\mathbf{q}_n\|$, $\|\dot{\mathbf{q}}_n\|$, and $\|\ddot{\mathbf{q}}_n\|$, respectively, both in the conservative system and the dissipative system, Eq. (A.31) with $C = 0.001$.

In this problem, the total linear momentum within a time step should not be conserved in the physical sense in the absence of the external load, any schemes never show $\mathbf{L}_n = \mathbf{L}_{n+1}$. This can be seen in Fig. 8.4 - Fig. 8.48, and we observe that the linear momentum is always bounded for all schemes both in the conservative and dissipative systems.

On the other hand, the total linear momentum within a time step should be conserved in the conservative system in the physical sense. The conservation of the angular momentum in $[t_n, t_{n+1}]$ in the sense of $\mathbf{J}_n = \mathbf{J}_{n+1}$ in the conservative system is observed only for some special cases. That is, the angular momentum is exactly conserved only for: $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\mathrm{s}} \in [0, 1]$ in the U0-based family or $\rho_\infty^{\min} = \rho_\infty^{\max} = \rho_\infty^{\mathrm{s}} = 1$ in the V0-based family in Algorithm 28 with $\eta_3 = 0$, $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\mathrm{s}} = 0$ in the U0-based family (which is equivalent to the central difference scheme) in Algorithm 29 with $\eta_3 = 0$, and $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\mathrm{s}} = 0$ in the U0-based family in Algorithm 30 with $\eta_3 = 0$. Note that we cannot obtain the exact angular momentum conserving scheme if we select $\eta_3 = 1$. The U0V0/V0U0 optimal-based schemes in Options I, II, and III are not angular momentum conserving schemes in general; however, they give better results in the qualitative sense by we selecting a high value of ρ_∞ close to but less than unity. It also can be improved by we selecting the lower time step size. In the dissipative system, all schemes, shown in Fig. 8.13 - Fig. 8.48, show that the angular momentum is always decaying both for $\eta_3 = 1$ and $\eta_3 = 0$, as expected. As can be seen from Fig. 8.4 - Fig. 8.12 in the conservative system, there exists no exact energy conserving explicit scheme in any options. The energy dissipation in the conservative system can be observed in the U0V0/V0U0 optimal-based schemes in any options, and we tend to have high energy dissipative features by selecting a lower value of ρ_∞. In the dissipative system, any schemes selected show the dissipative behaviors of the mechanical energy; see Fig. 8.13 - Fig. 8.48 both for $\eta_3 = 1$ and $\eta_3 = 0$. The initial conditions used for the simulation were:

$$\begin{aligned} \mathbf{q}(t_0) &= \mathbf{q}_0 = (0.1, 1.0)^T \\ \dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0 = (0.1, 0.1)^T \end{aligned} \tag{8.34}$$

The final time of the simulation are selected to be $T = 30$ sec.

3D Kepler's Problem: In this example problem, we applied the same selected schemes from the previous example problem. And we observed similar numerical behaviors of the algorithms in the time accuracies and the linear and angular momenta and mechanical energy conservations. The second-order time accuracies

in the configuration, velocity, and acceleration in the forced-damped system, with $C = 0.01$ and the external loading $\mathbf{f}^{\text{ext}} = \bar{\mathbf{f}} \sin(t)$ where $\bar{\mathbf{f}} = (0.1, 0.0, 0.0)^T$, with the selected schemes are observed; see Fig. 8.49, Fig. 8.50, and Fig. 8.51 for Option I, II, and III, respectively, with $\eta_3 = 1$; and see Fig. 8.61, Fig. 8.62, and Fig. 8.63 for Option I, II, and III, respectively, with $\eta_3 = 0$. Similar to the nonlinear oscillator problem presented above, the total linear momentum in the absence of the external loading should not be onserved in the physical sense in this problem. The angular momentum is exactly conserved only for: $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\text{s}} \in [0, 1]$ in the U0-based family or $\rho_\infty^{\min} = \rho_\infty^{\max} = \rho_\infty^{\text{s}} = 1$ in the V0-based family in Algorithm 28 with $\eta_3 = 0$, $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\text{s}} = 0$ in the U0-based family (which is equivalent to the central difference scheme) in Algorithm 29 with $\eta_3 = 0$, and $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\text{s}} = 0$ in the U0-based family in Algorithm 30 with $\eta_3 = 0$. Note that we cannot obtain the exact angular momentum conserving scheme if we select $\eta_3 = 1$. And there exists no exact energy conserving scheme as expected. The time histories of the mechanical energy, linear and angular momenta, configuration, velocity, and acceleration in the conservative system are shown in Fig. 8.52 - Fig. 8.60 for $\eta_3 = 1$ and Fig. 8.61 - Fig. 8.75 for $\eta_3 = 0$. The initial conditions used for the simulation were:

$$\begin{aligned} \mathbf{q}(t_0) &= \mathbf{q}_0 = (0.9/\sqrt{2}, 0.0, 0.9/\sqrt{2})^T \\ \dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0 = (0.0, -100/9, 0.0)^T \end{aligned} \tag{8.35}$$

The final time of the simulation are selected to be $T = 20$ sec.

Lennard-Jones potential problem: Again, in this example problem, we applied the same selected schemes from the previous example problem. And we basically observed the same numerical behaviors of the algorithms in the time accuracies and the angular momenta and mechanical energy conservations. But, note that the total linear momentum should be conserved exactly in the sense of $\mathbf{L}_n = \mathbf{L}_{n+1}$ in the physical sense in this problem in the absence of the external loading. The second-order time accuracies in the configuration, velocity, and acceleration in the forced-damped system, with $C = 0.01$ and the external loading $\mathbf{f}^{\text{ext}} = \bar{\mathbf{f}} \sin(t)$ where $\bar{\mathbf{f}} = (0.1, 0.0, 0.0, 0.0, -0.5, 0.0)^T$, with the selected schemes

are observed; see Fig. 8.76, Fig. 8.77, and Fig. 8.78 for Option I, II, and III, respectively, with $\eta_3 = 1$; and see Fig. 8.88, Fig. 8.89, and Fig. 8.90 for Option I, II, and III, respectively, with $\eta_3 = 0$. The linear momentum is exactly conserved for any spectral conditions in any options for both $\eta_3 = 1$ and $\eta_3 = 0$; see Fig. 8.79 - Fig. 8.102. The angular momentum is exactly conserved only for: $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\mathrm{s}} \in [0, 1]$ in the U0-based family or $\rho_\infty^{\min} = \rho_\infty^{\max} = \rho_\infty^{\mathrm{s}} = 1$ in the V0-based family in Algorithm 28 with $\eta_3 = 0$, $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\mathrm{s}} = 0$ in the U0-based family (which is equivalent to the central difference scheme) in Algorithm 29 with $\eta_3 = 0$, and $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\mathrm{s}} = 0$ in the U0-based family in Algorithm 30 with $\eta_3 = 0$. Note that we cannot obtain the exact angular momentum conserving scheme if we select $\eta_3 = 1$. And there exists no exact energy conserving scheme as expected. The time histories of the mechanical energy, linear and angular momenta, configuration, velocity, and acceleration in the conservative system are shown in Fig. 8.79 - Fig. 8.87 for $\eta_3 = 1$ and Fig. 8.88 - Fig. 8.102 for $\eta_3 = 0$. The initial conditions used for the simulation were:

$$\begin{aligned} \mathbf{q}(t_0) &= \mathbf{q}_0 = (0.0, -0.5, 0.0, 0.0, 0.5, 0.0)^T \\ \dot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}_0 = (5.0, 1.0, 2.0, 10.0, 3.0, -1.0)^T \end{aligned} \tag{8.36}$$

The final time of the simulation are selected to be $T = 2$ sec.

Spring-Pendulum problem: In addition, in order to emphasize that the order of time accuracy of the algorithms remains to be two for the implicit and explicit treatments in the general case, the accuracy plots for the spring-pendulum problem, simulated by the PCE-GSSSS algorithms with $\eta_3 = 0$ are shown in Fig. 8.103 for the implicit treatment of the velocity term and in Fig. 8.104 for the explicit treatment of the velocity term [legends are: the configuration ($\square$), velocity ($\triangle$), and acceleration ($\circ$)]. The input parameters are: $g = 9.81$, $L_0 = 10$, $k = 100$, $r_0 = 0.3$, and $\theta_0 = 50(\pi/180)$.

Here is the summary of the numerical results:

Remark 31

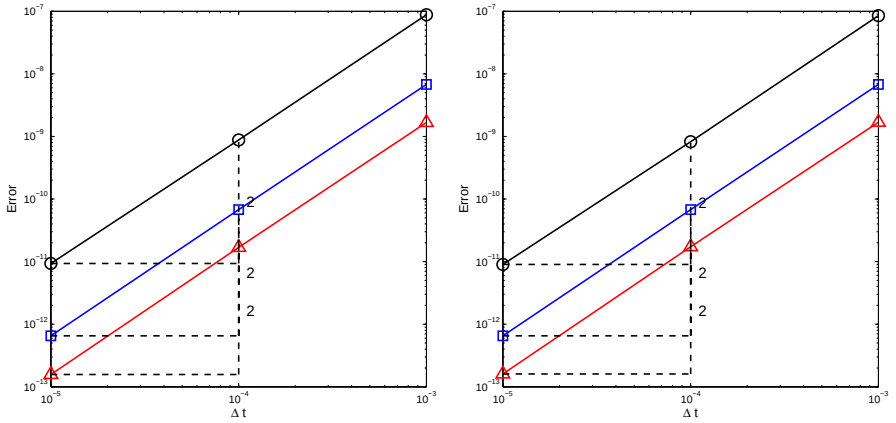
1. *The order of time accuracies in the configuration, velocity, and acceleration is 2 for any selections of the spectral conditions and η_i (for $i = 1, 2, 3$).*

2. *There is no energy conserving explicit scheme both for* $\eta_3 = 1$ *and* $\eta_3 = 0$ *in the conservative system.*

3. *There is no angular momentum conserving explicit scheme for* $\eta_3 = 1$ *in the conservative system.*

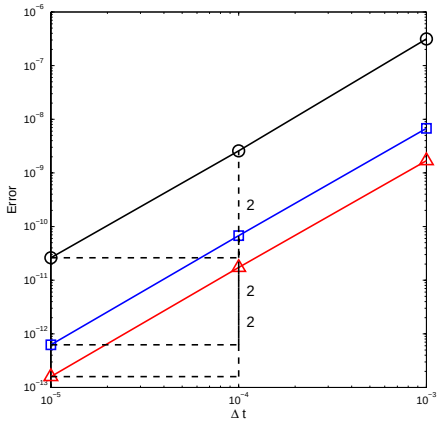
4. *The angular momentum conserving schemes, these schemes are variational actually, are obtained by selecting the following spectral conditions with* $\eta_3 = 0$ *in the conservative system:*
   Option I: $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ *and* $\rho_\infty^{\mathrm{s}} \in [0, 1]$ *in the U0-based family or* $\rho_\infty^{\min} = \rho_\infty^{\max} = \rho_\infty^{\mathrm{s}} = 1$ *in the V0-based family*
   Option II: $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ *and* $\rho_\infty^{\mathrm{s}} = 0$ *in the U0-based family (Central difference scheme)*
   Option III: $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ *and* $\rho_\infty^{\mathrm{s}} = 0$ *in the U0-based family*

5. *In the dissipative system, all schemes show the energy and angular momentum dissipative features.*

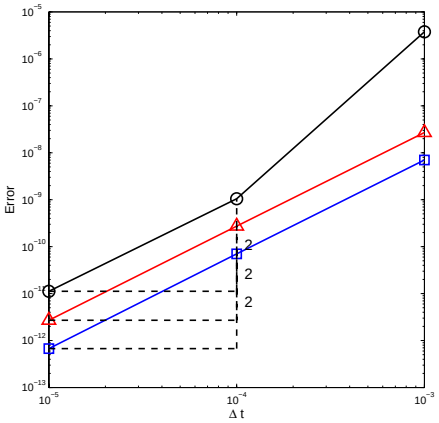


(a) U0V0/V0U0(1.0,1.0,1.0)

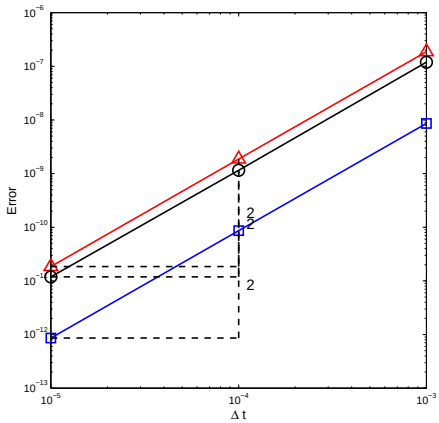
(b) U0V0(1.0,1.0,0.0)



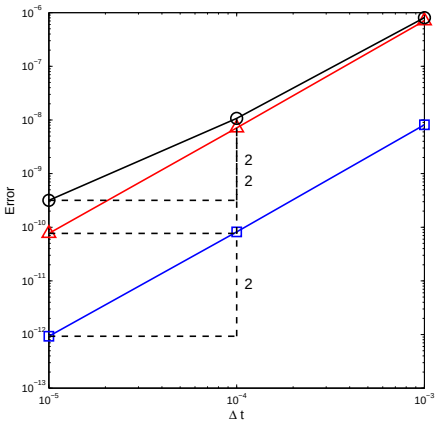
(c) V0U0(1.0,1.0,0.0)



(d) U0V0/V0U0(0.8,1.0,0.8)

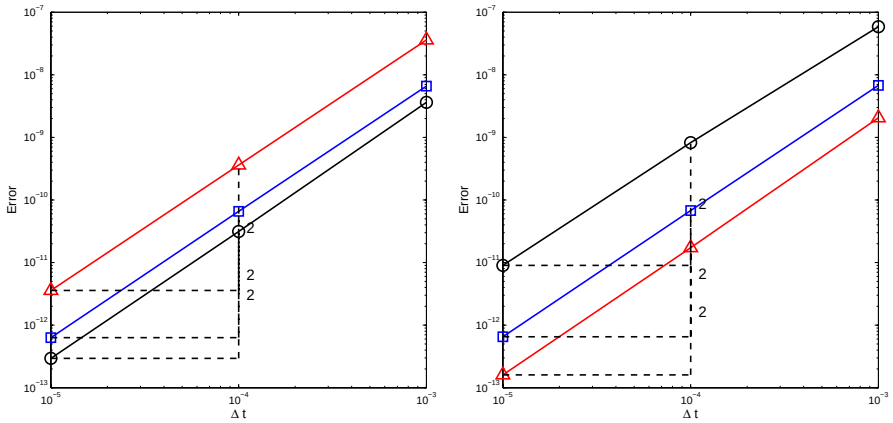


(e) U0V1(0.6,0.9,0.2)



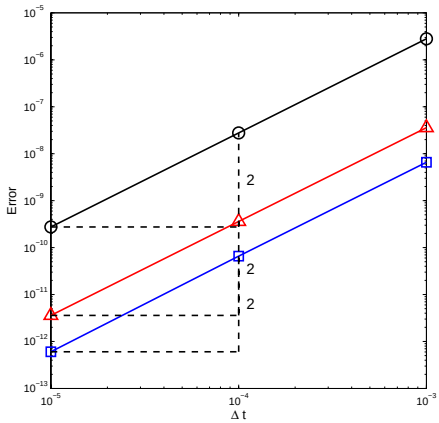
(f) V0U1(0.6,0.9,0.2)

Figure 8.1: Time accuracies in the configuration (□), velocity (△), and acceleration (○) in the forced mechanical system. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 1$ (Option I)]

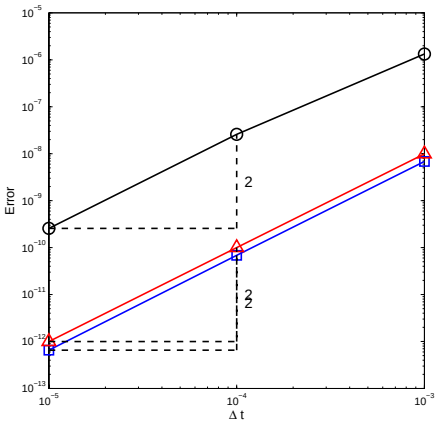


(a) U0V0/V0U0(1.0,1.0,1.0)

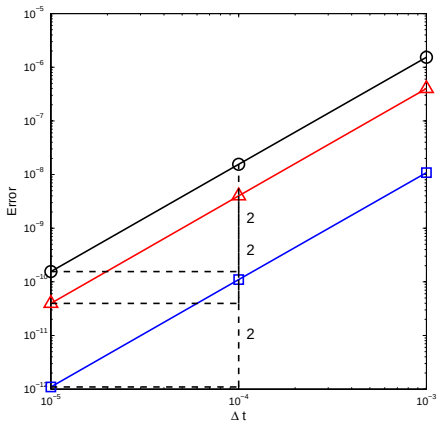
(b) U0V0(1.0,1.0,0.0)



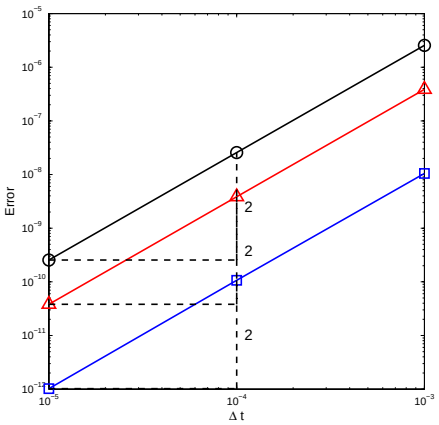
(c) V0U0(1.0,1.0,0.0)



(d) U0V0/V0U0(0.8,1.0,0.8)

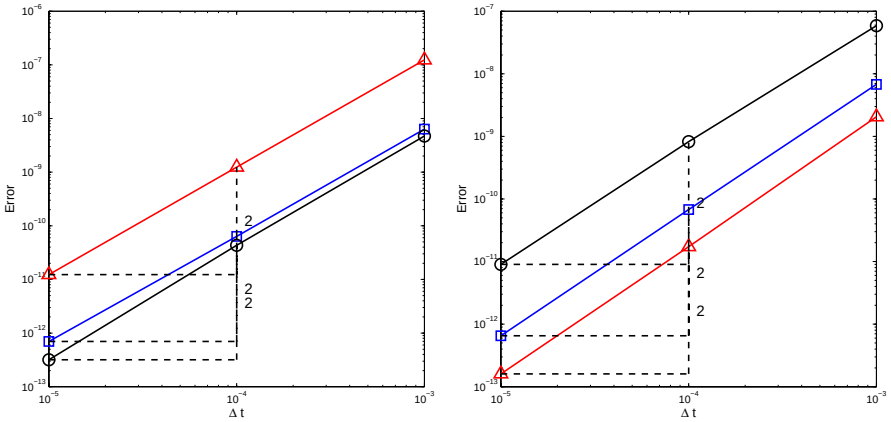


(e) U0V1(0.6,0.9,0.2)



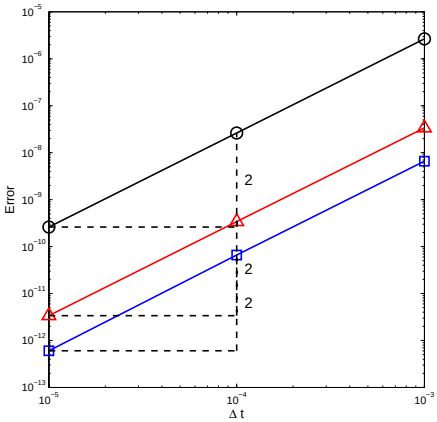
(f) V0U1(0.6,0.9,0.2)

Figure 8.2: Time accuracies in the configuration (□), velocity (△), and acceleration (○) in the forced mechani [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 1$ (Option II)]

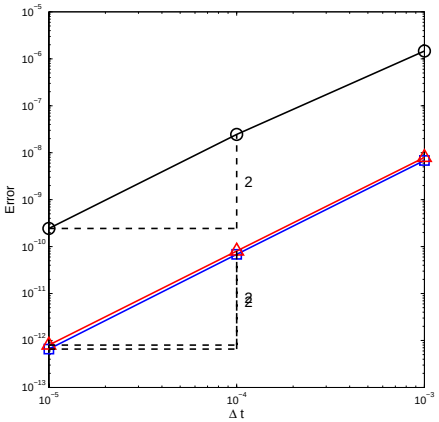


(a) U0V0/V0U0(1.0,1.0,1.0)

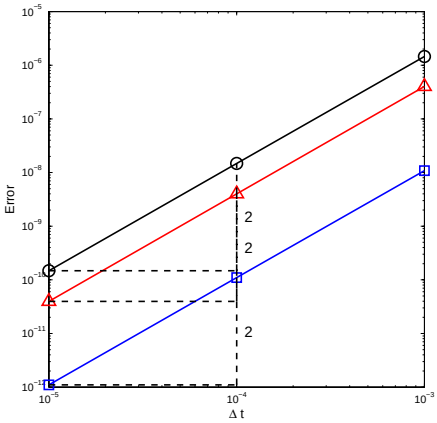
(b) U0V0(1.0,1.0,0.0)



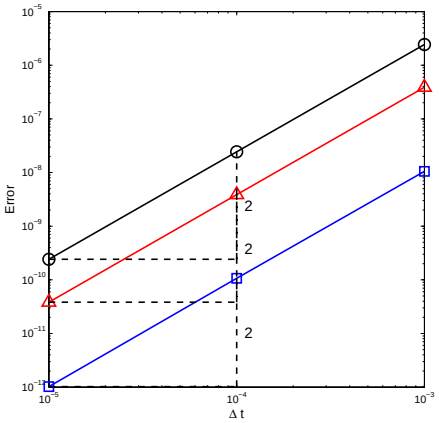
(c) V0U0(1.0,1.0,0.0)



(d) U0V0/V0U0(0.8,1.0,0.8)

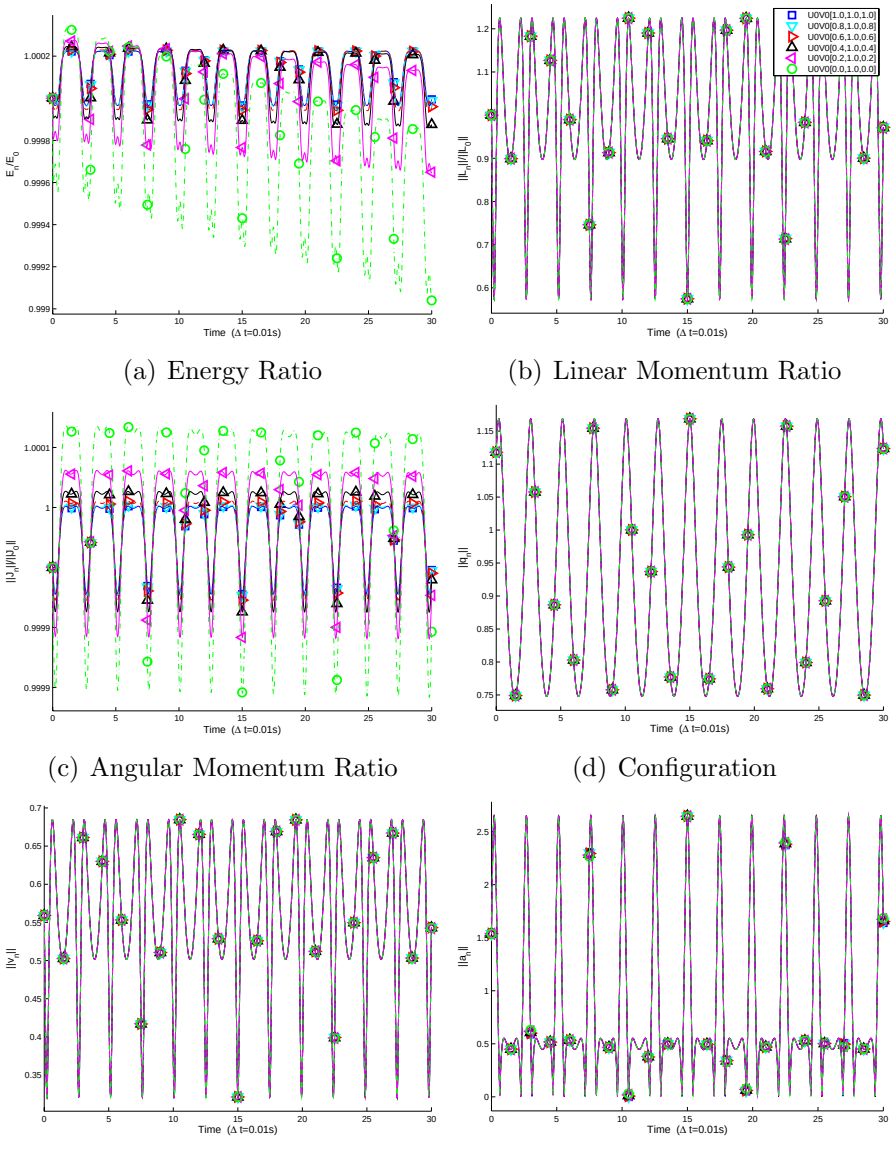


(e) U0V1(0.6,0.9,0.2)



(f) V0U1(0.6,0.9,0.2)

Figure 8.3: Time accuracies in the configuration, velocity, and acceleration in the forced mechanical system. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 1$ (Option III)]

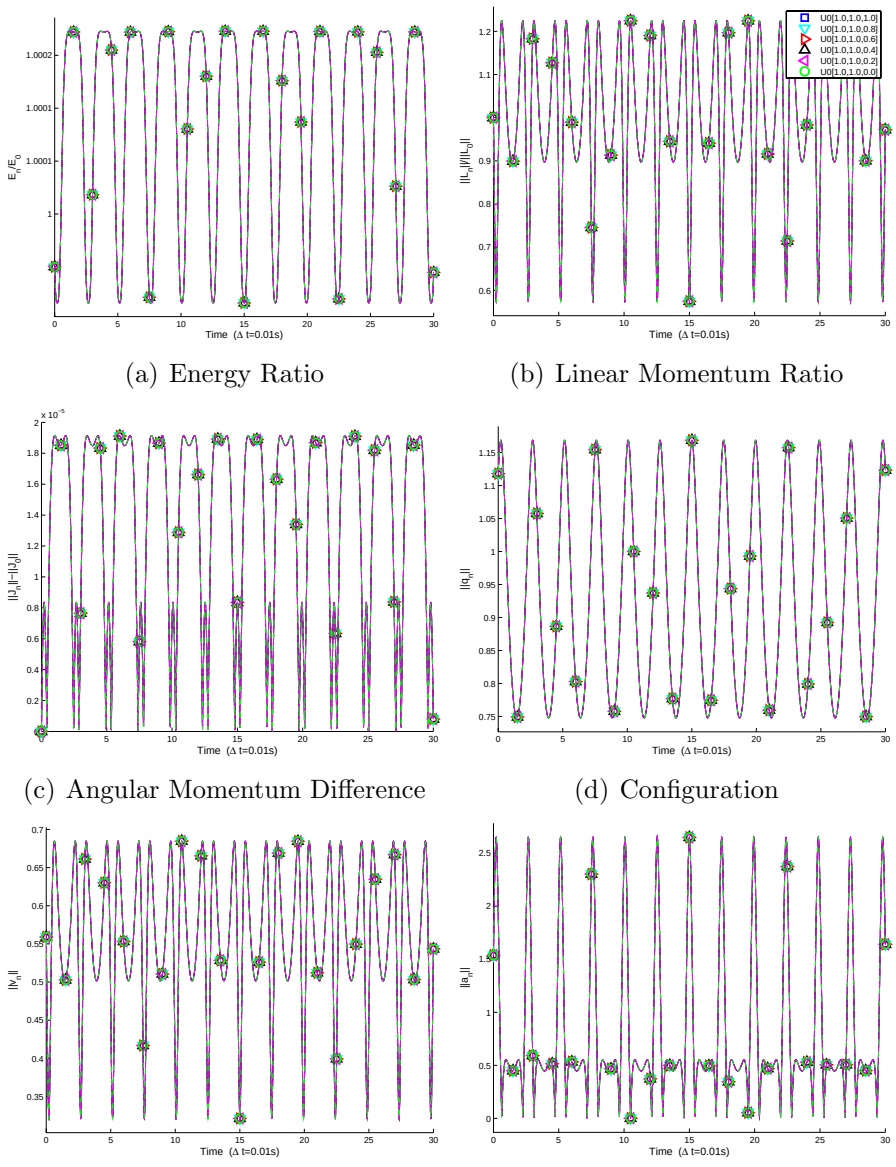


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.4: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option I) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]

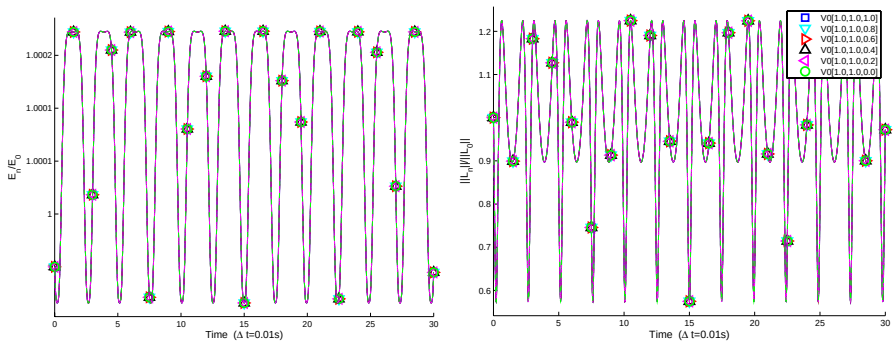


(a) Energy Ratio (b) Linear Momentum Ratio

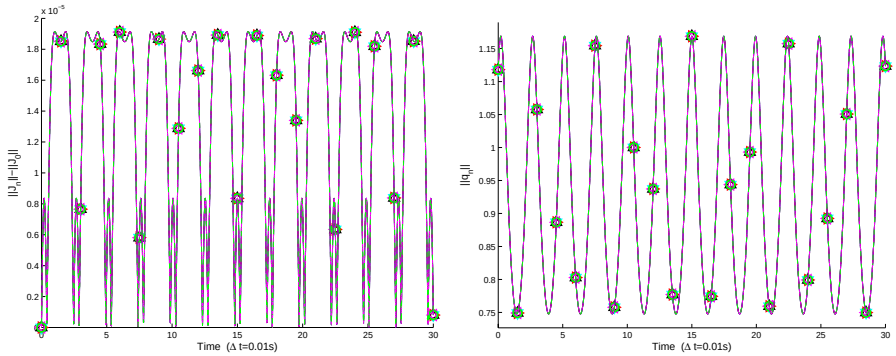
(c) Angular Momentum Difference (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

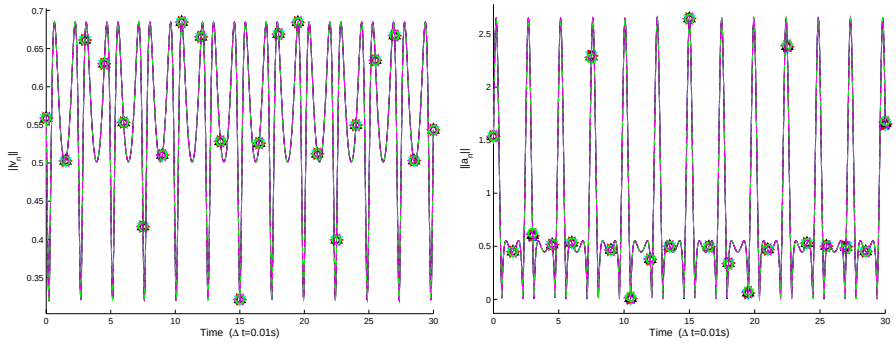
Figure 8.5: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option I) - U0V0(1.0,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum Ratio

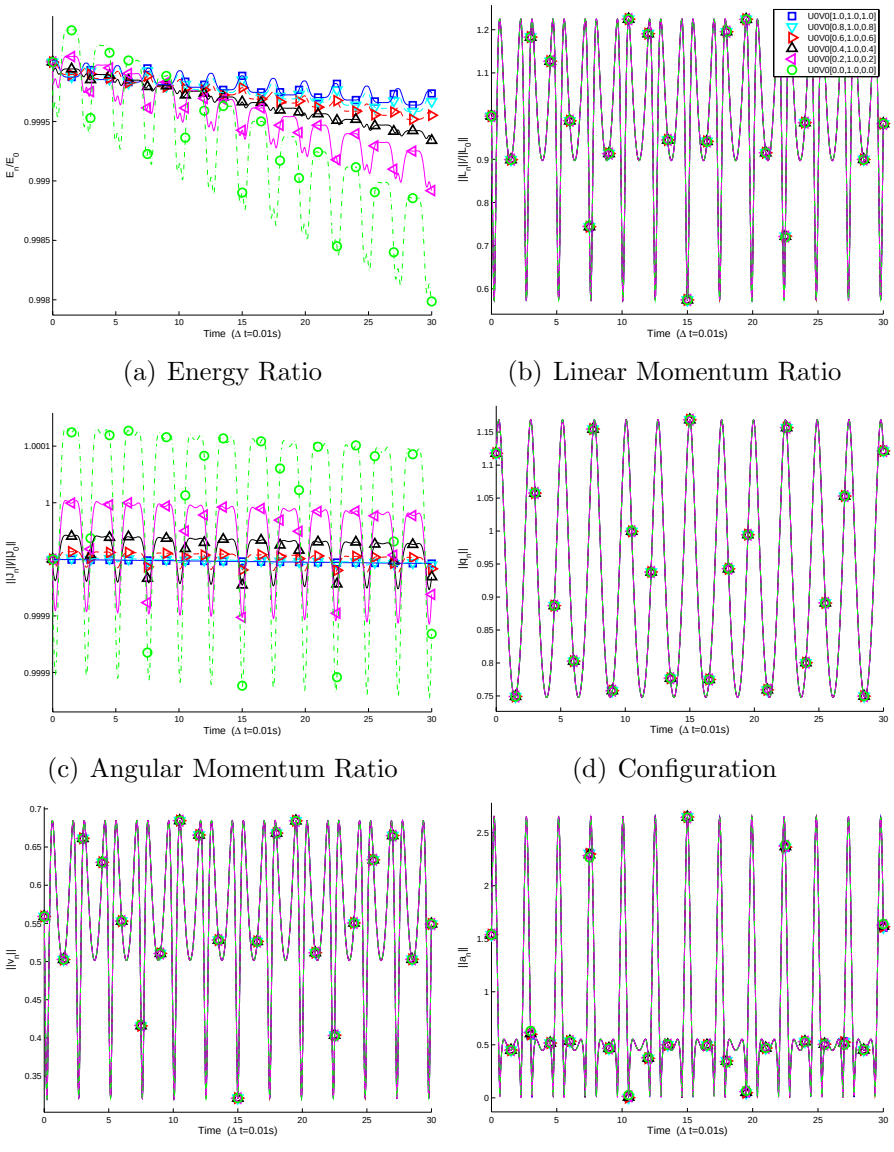


(c) Angular Momentum Difference (d) Configuration



(e) Velocity Norm (f) Acceleration Norm

Figure 8.6: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option I) - V0U0(1.0,1.0,ρ_∞)]

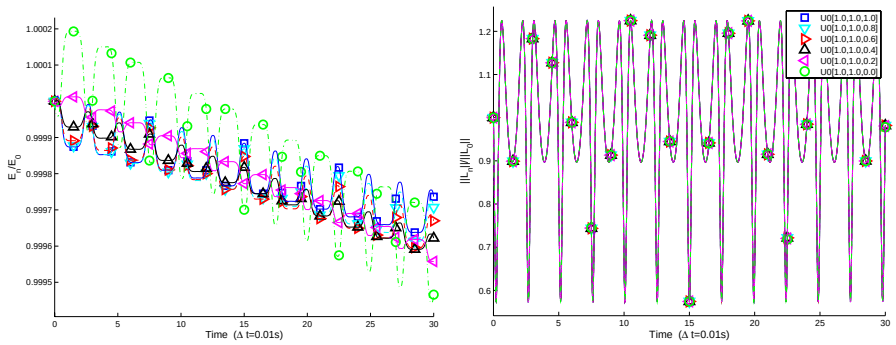


(a) Energy Ratio (b) Linear Momentum Ratio

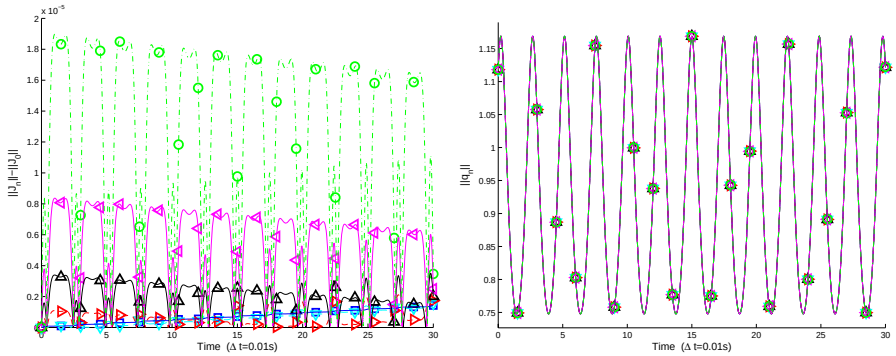
(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

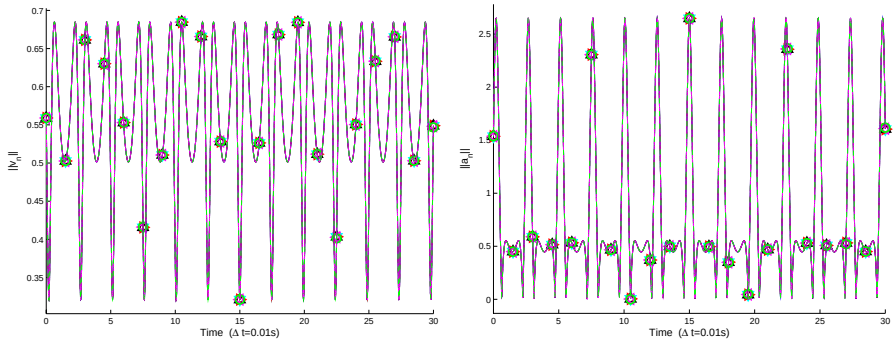
Figure 8.7: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option II) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum Ratio

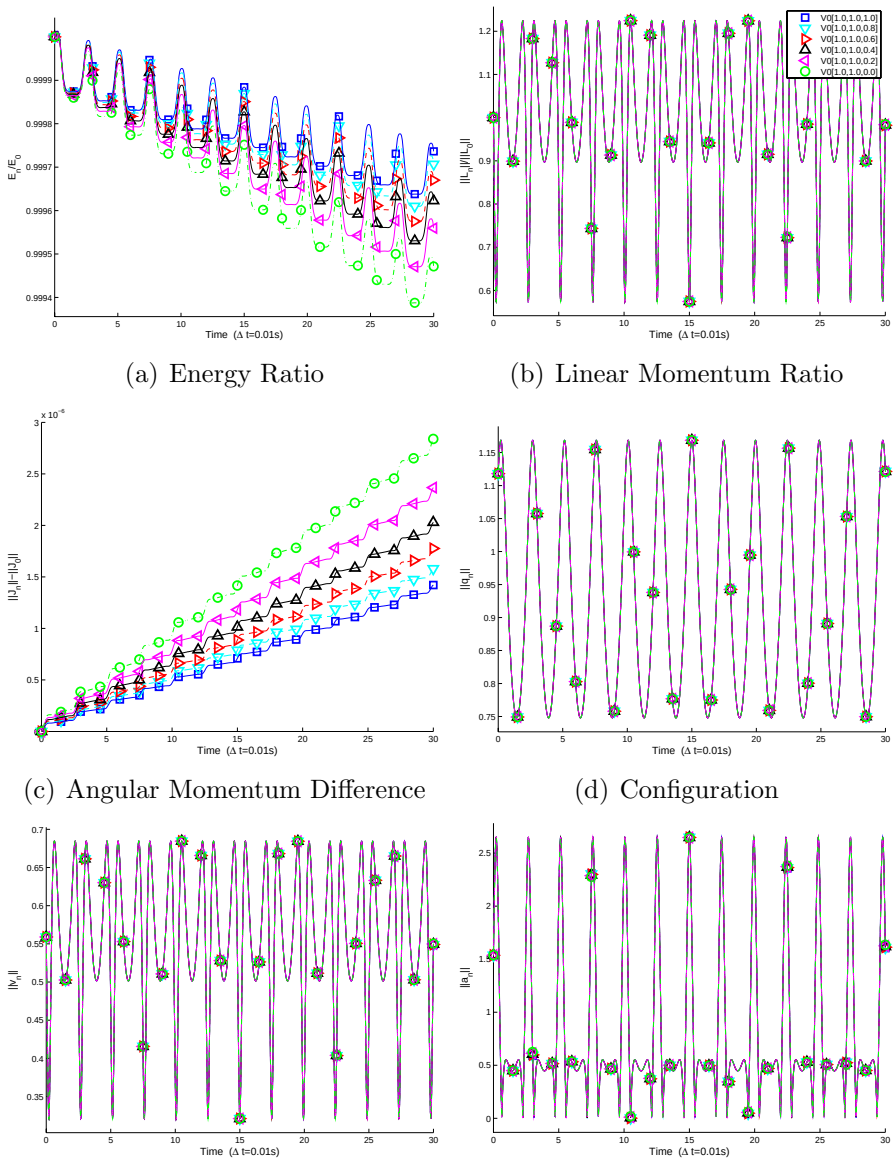


(c) Angular Momentum Difference (d) Configuration



(e) Velocity Norm (f) Acceleration Norm

Figure 8.8: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option II) - U0V0(1.0,1.0,ρ_∞)]



(a) Energy Ratio

(b) Linear Momentum Ratio

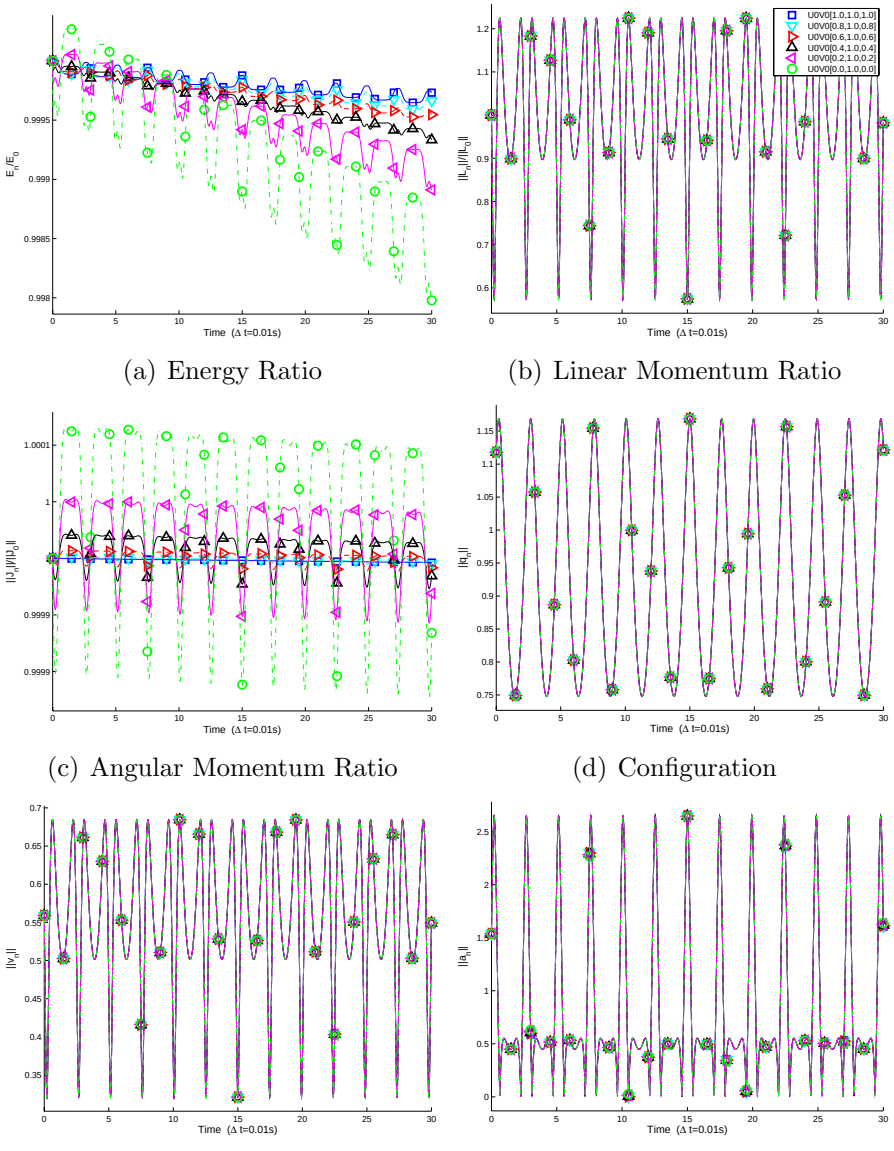
(c) Angular Momentum Difference

(d) Configuration

(e) Velocity Norm

(f) Acceleration Norm

Figure 8.9: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option II) - V0U0(1.0,1.0,ρ_∞)]

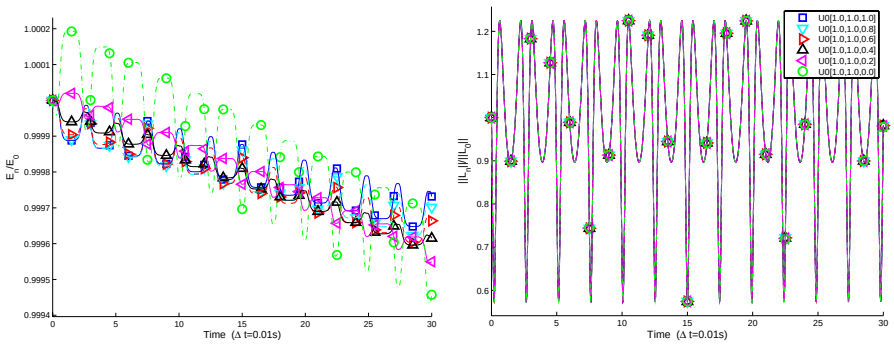


(a) Energy Ratio (b) Linear Momentum Ratio

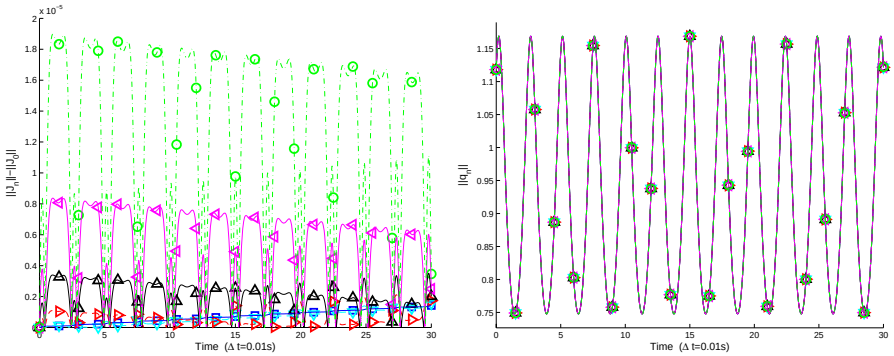
(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

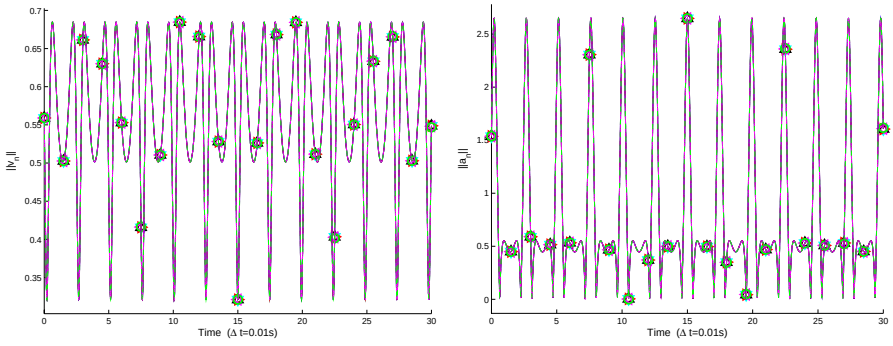
Figure 8.10: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option III) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum Ratio

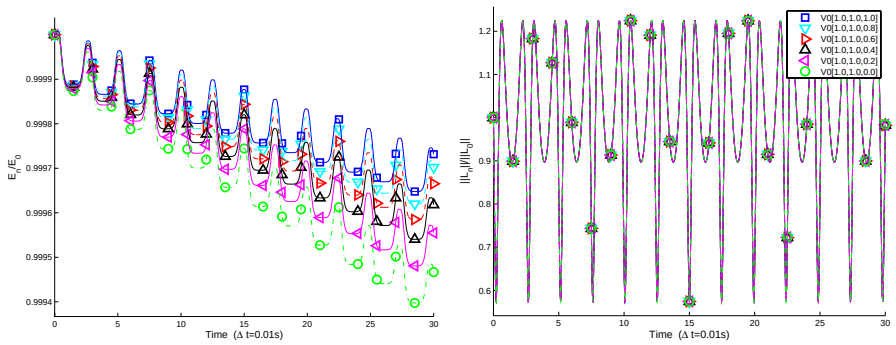


(c) Angular Momentum Difference (d) Configuration

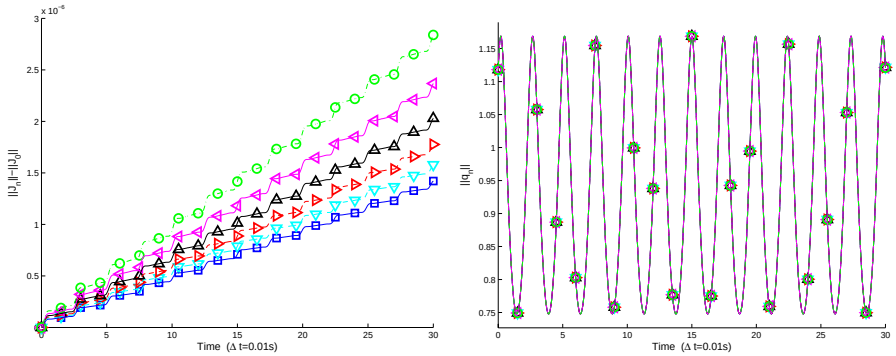


(e) Velocity Norm (f) Acceleration Norm

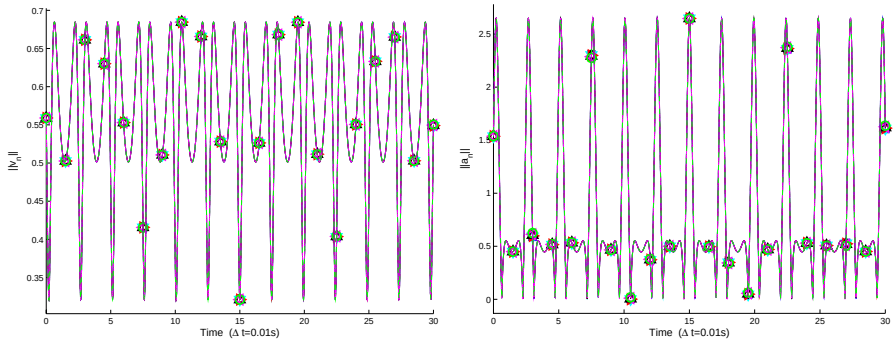
Figure 8.11: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option III) - U0V0(1.0,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum Ratio

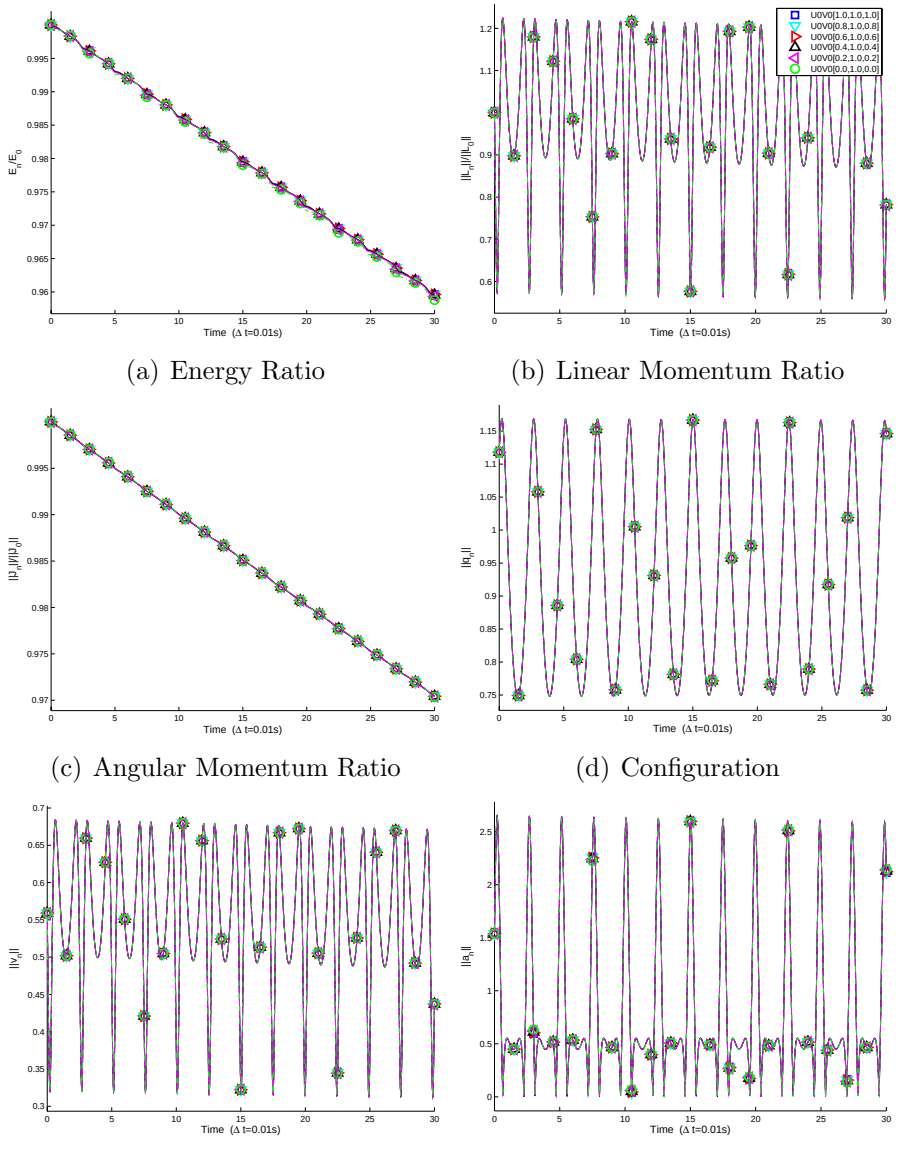


(c) Angular Momentum Difference (d) Configuration



(e) Velocity Norm (f) Acceleration Norm

Figure 8.12: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option III) - V0U0(1.0,1.0,ρ_∞)]

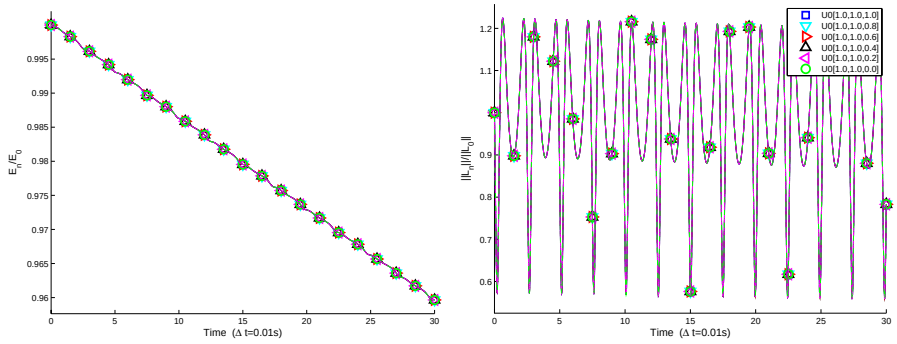


(a) Energy Ratio (b) Linear Momentum Ratio

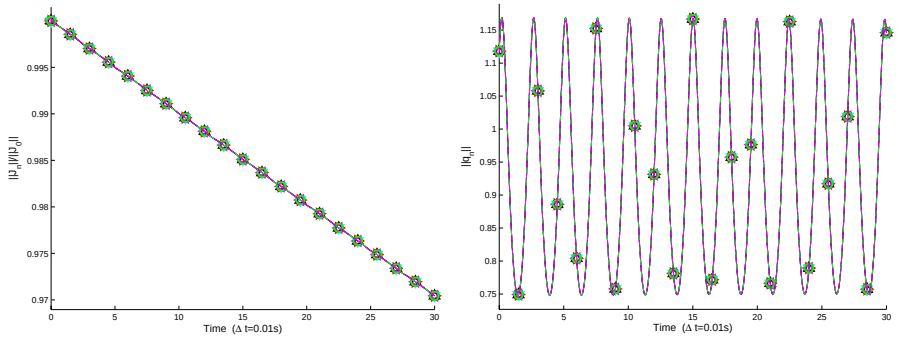
(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

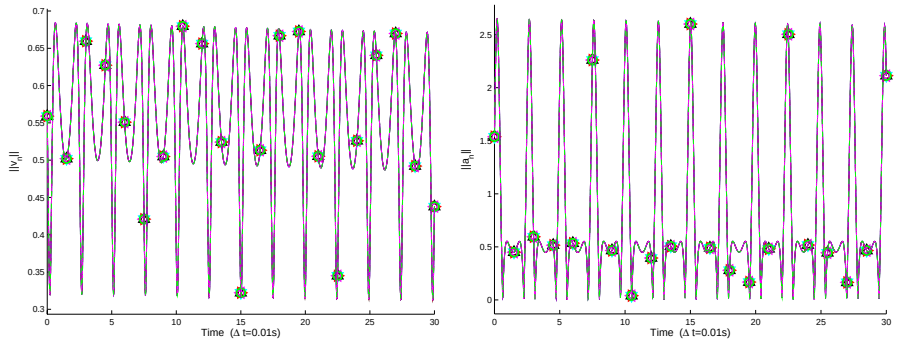
Figure 8.13: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 1$ (Option I) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum Ratio

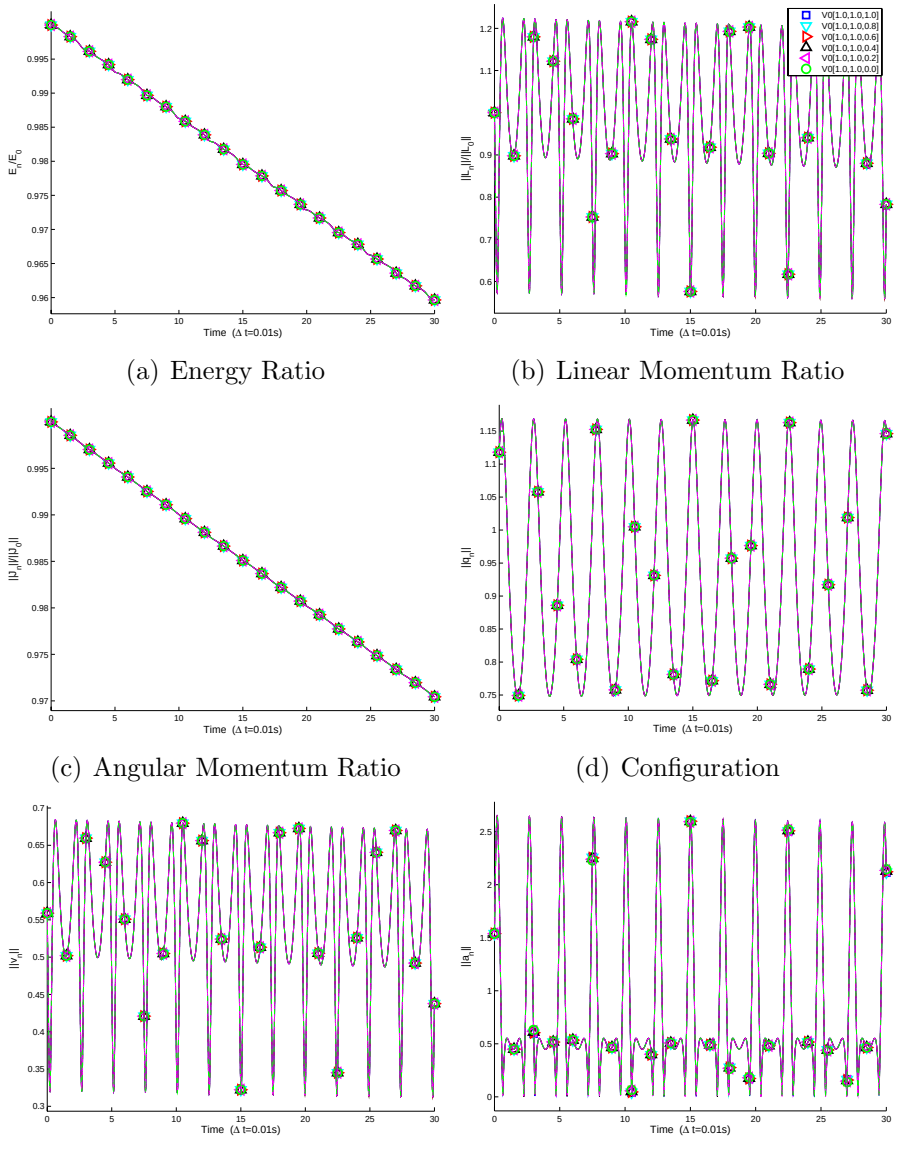


(c) Angular Momentum Ratio (d) Configuration



(e) Velocity Norm (f) Acceleration Norm

Figure 8.14: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 1$ (Option I) - U0V0(1.0,1.0,ρ_∞)]



(a) Energy Ratio

(b) Linear Momentum Ratio

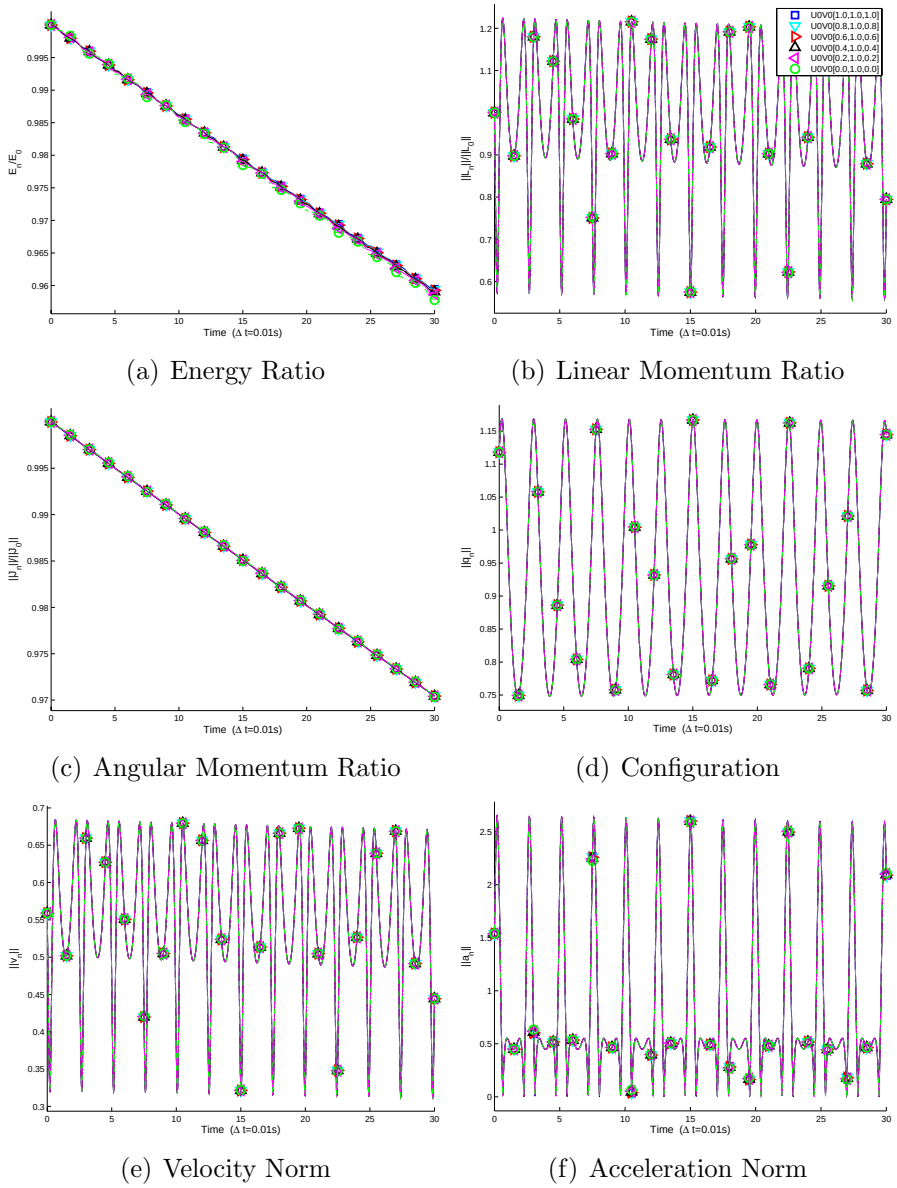
(c) Angular Momentum Ratio

(d) Configuration

(e) Velocity Norm

(f) Acceleration Norm

Figure 8.15: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 1$ (Option I) - V0U0(1.0,1.0,ρ_∞)]



(a) Energy Ratio

(b) Linear Momentum Ratio

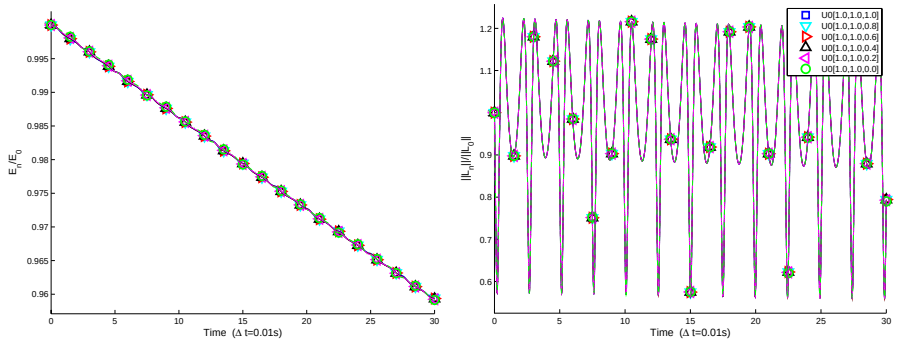
(c) Angular Momentum Ratio

(d) Configuration

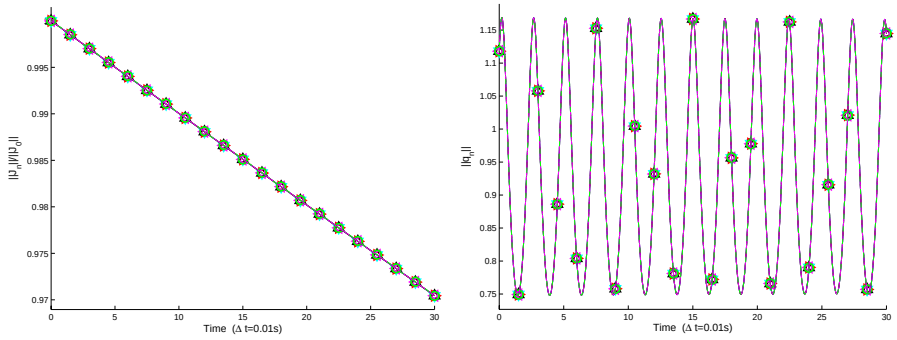
(e) Velocity Norm

(f) Acceleration Norm

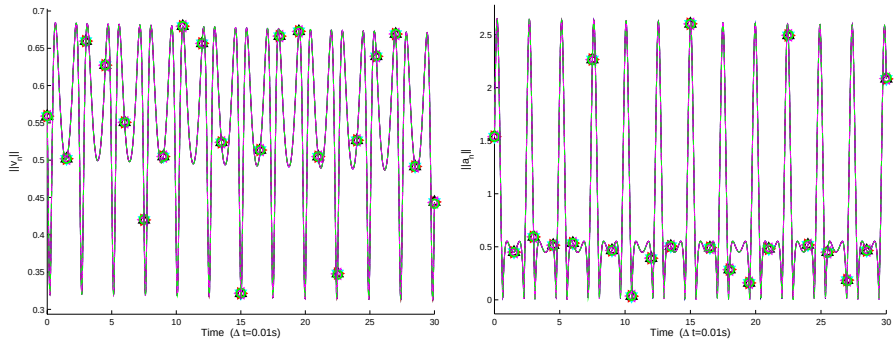
Figure 8.16: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 1$ (Option II) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum Ratio

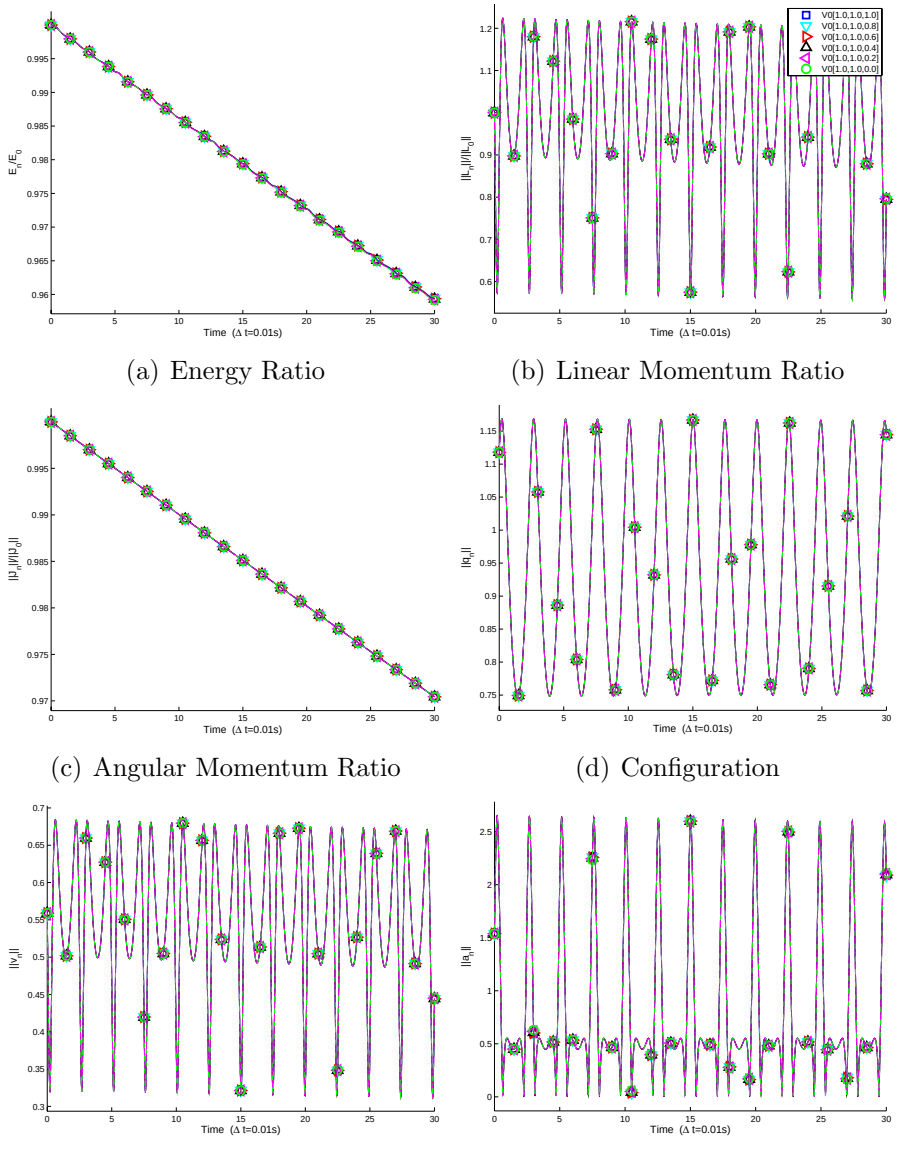


(c) Angular Momentum Ratio (d) Configuration



(e) Velocity Norm (f) Acceleration Norm

Figure 8.17: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 1$ (Option II) - U0V0(1.0,1.0,ρ_∞)]

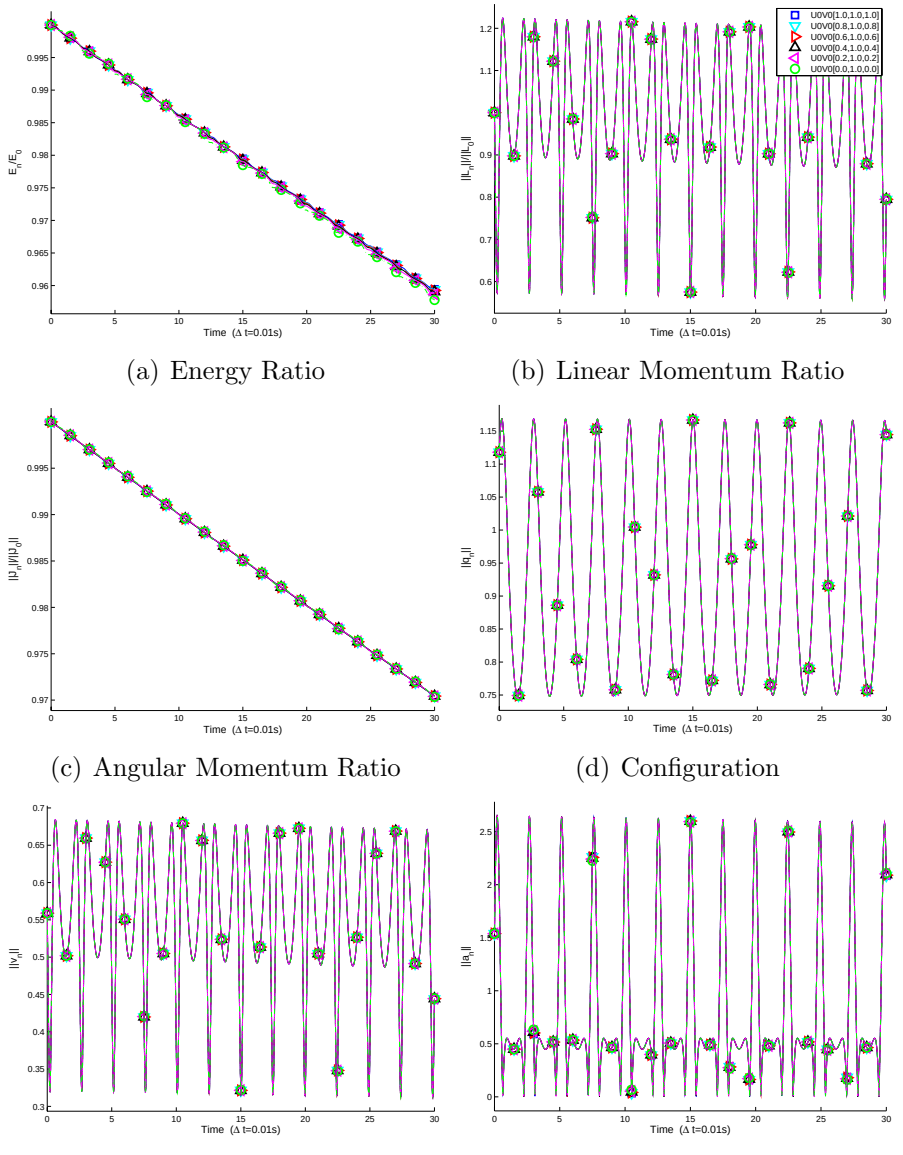


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.18: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 1$ (Option II) - V0U0(1.0,1.0,ρ_∞)]

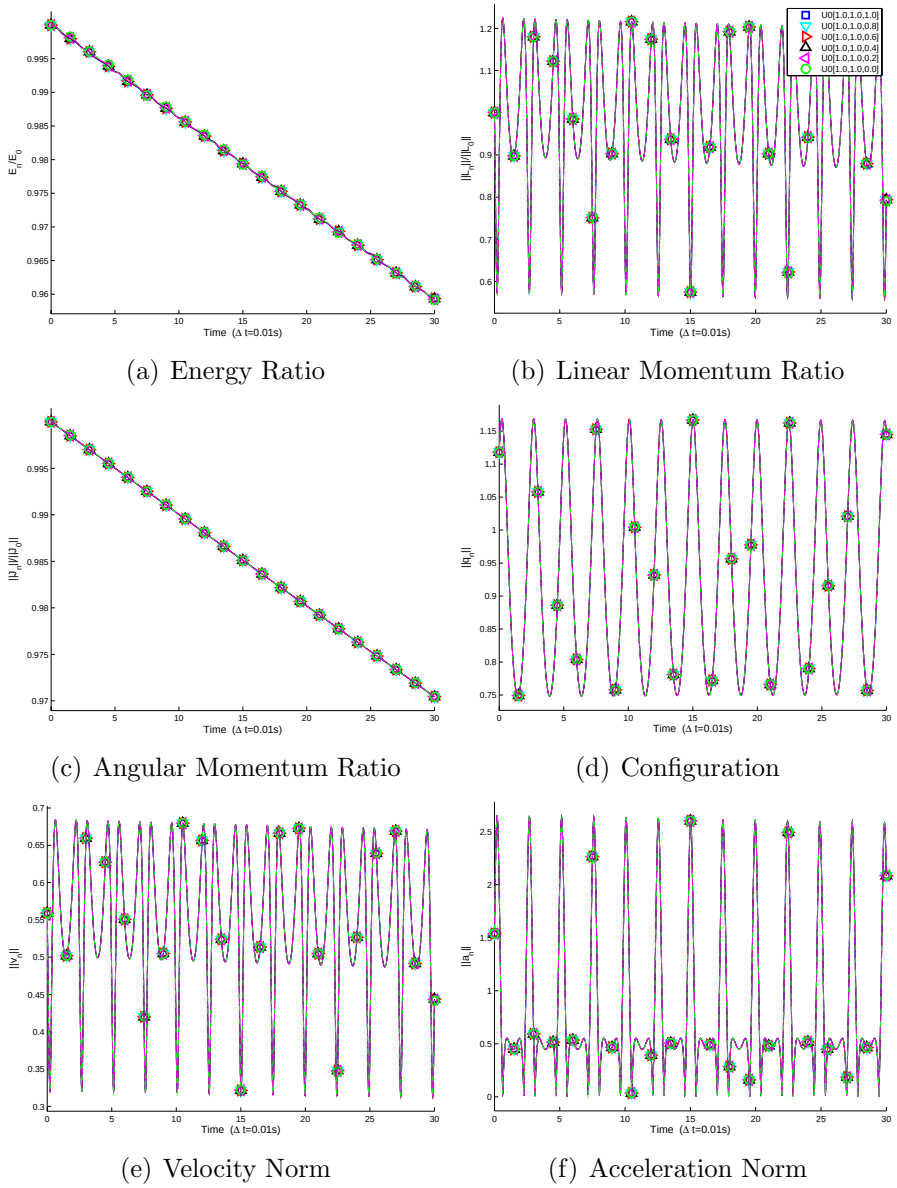


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.19: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 1$ (Option III) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Energy Ratio

(b) Linear Momentum Ratio

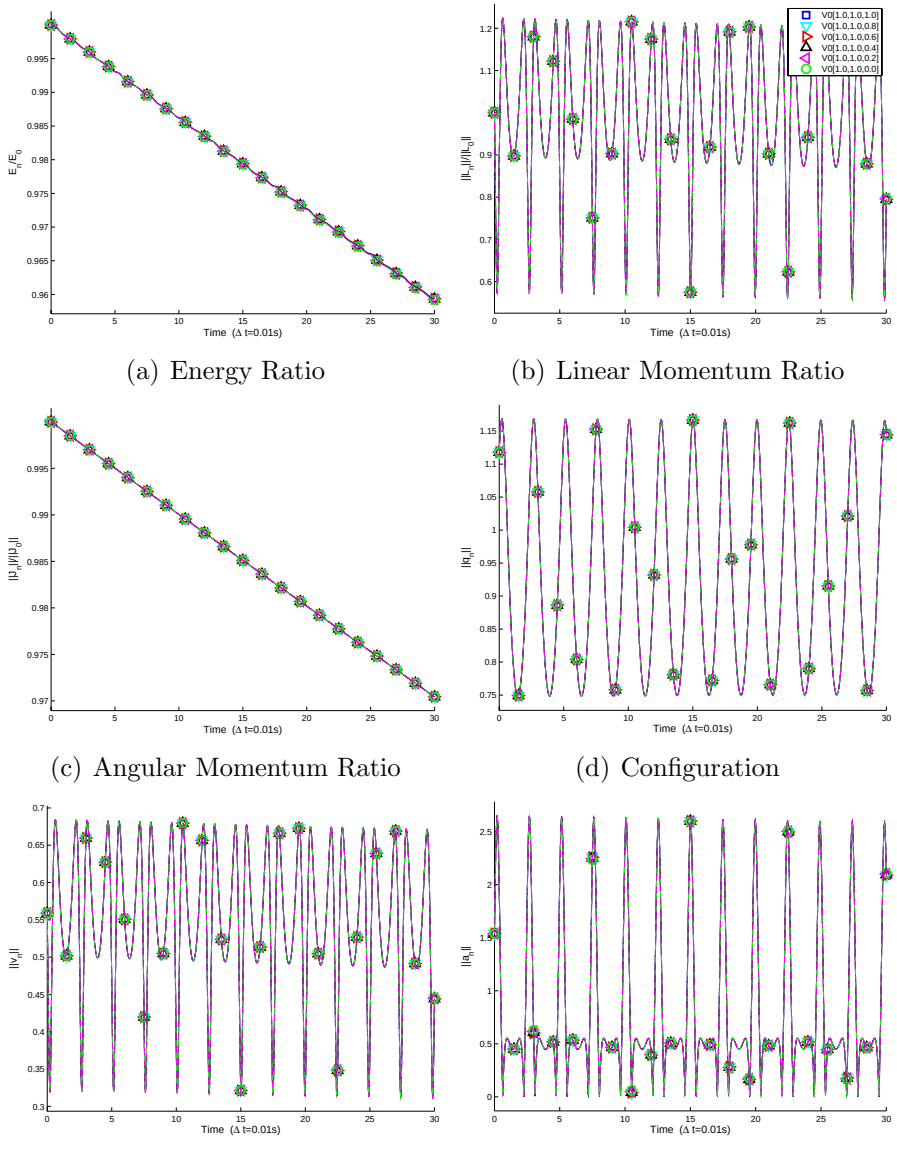
(c) Angular Momentum Ratio

(d) Configuration

(e) Velocity Norm

(f) Acceleration Norm

Figure 8.20: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 1$ (Option III) - U0V0(1.0,1.0,ρ_∞)]

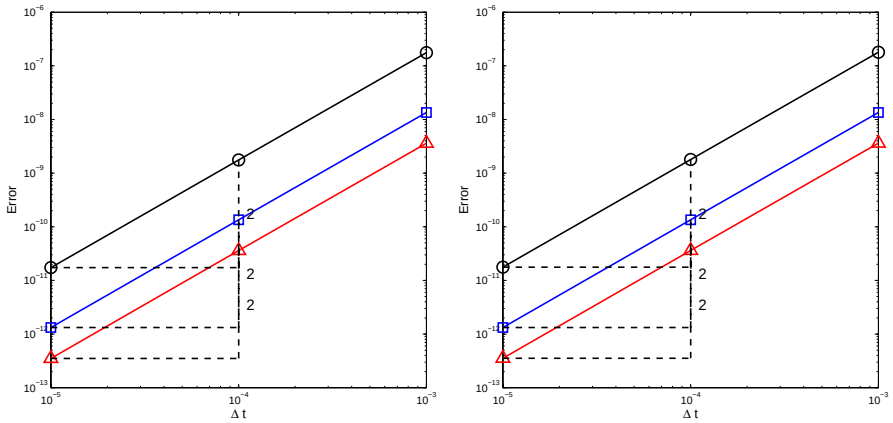


(a) Energy Ratio (b) Linear Momentum Ratio

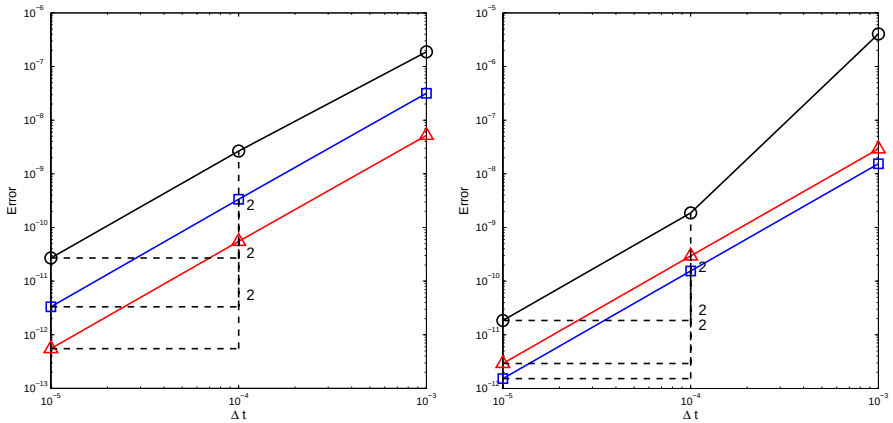
(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

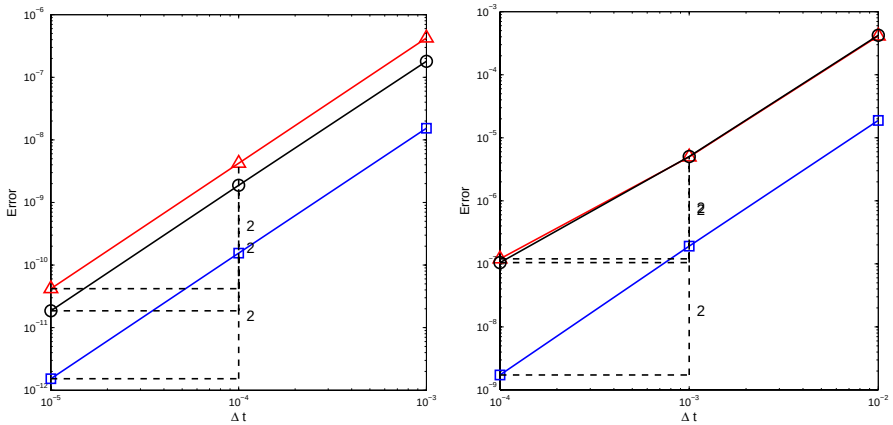
Figure 8.21: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 1$ (Option III) - V0U0(1.0,1.0,ρ_∞)]



(a) U0V0/V0U0(1.0,1.0,1.0) (b) U0V0(1.0,1.0,0.0)

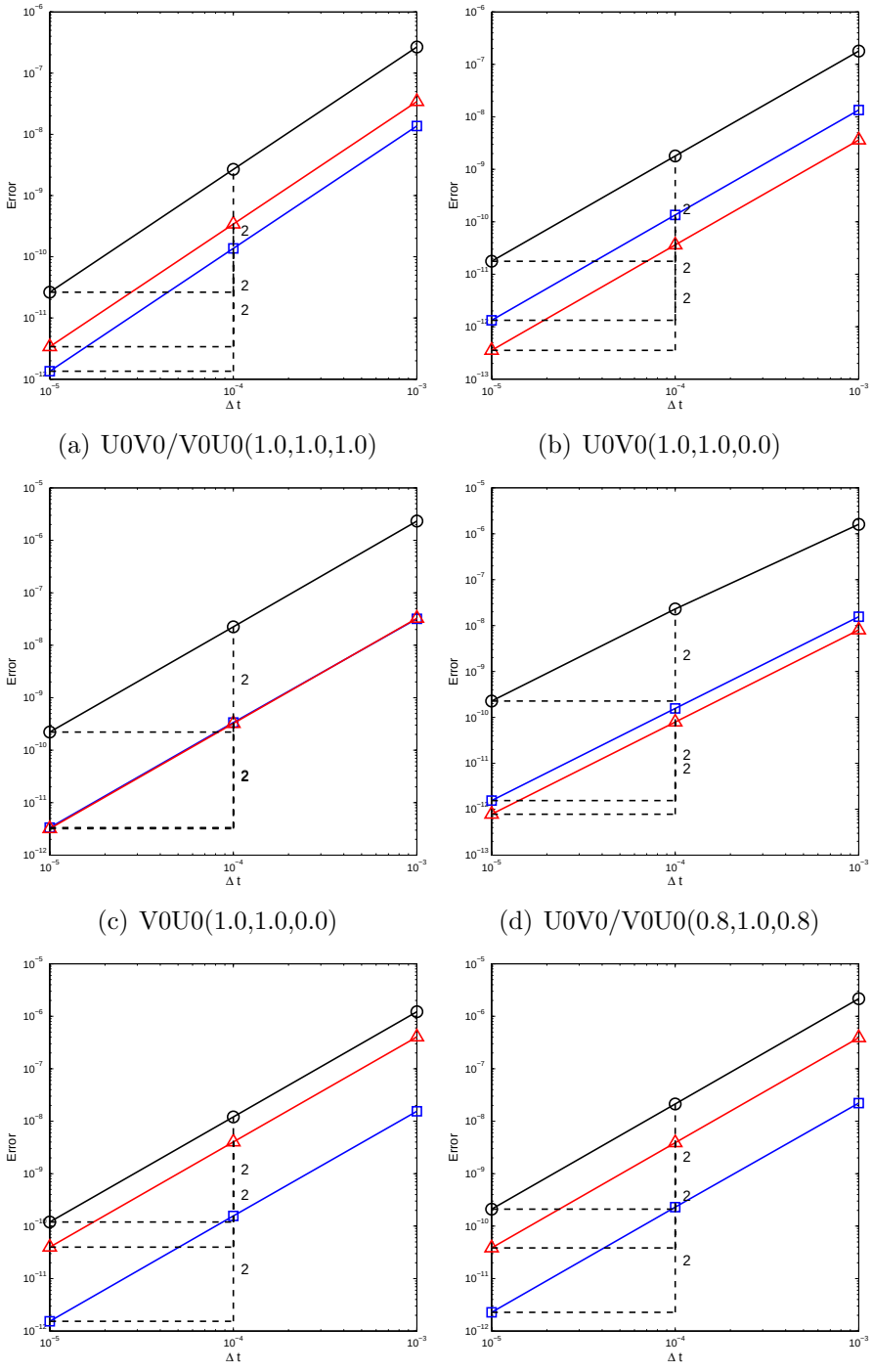


(c) V0U0(1.0,1.0,0.0) (d) U0V0/V0U0(0.8,1.0,0.8)



(e) U0V1(0.6,0.9,0.2) (f) V0U1(0.6,0.9,0.2)

Figure 8.22: Time accuracies in the configuration (□), velocity (△), and acceleration (○) in the forced mechanical system. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 0$ (Option I)]



(a) U0V0/V0U0(1.0,1.0,1.0)

(b) U0V0(1.0,1.0,0.0)

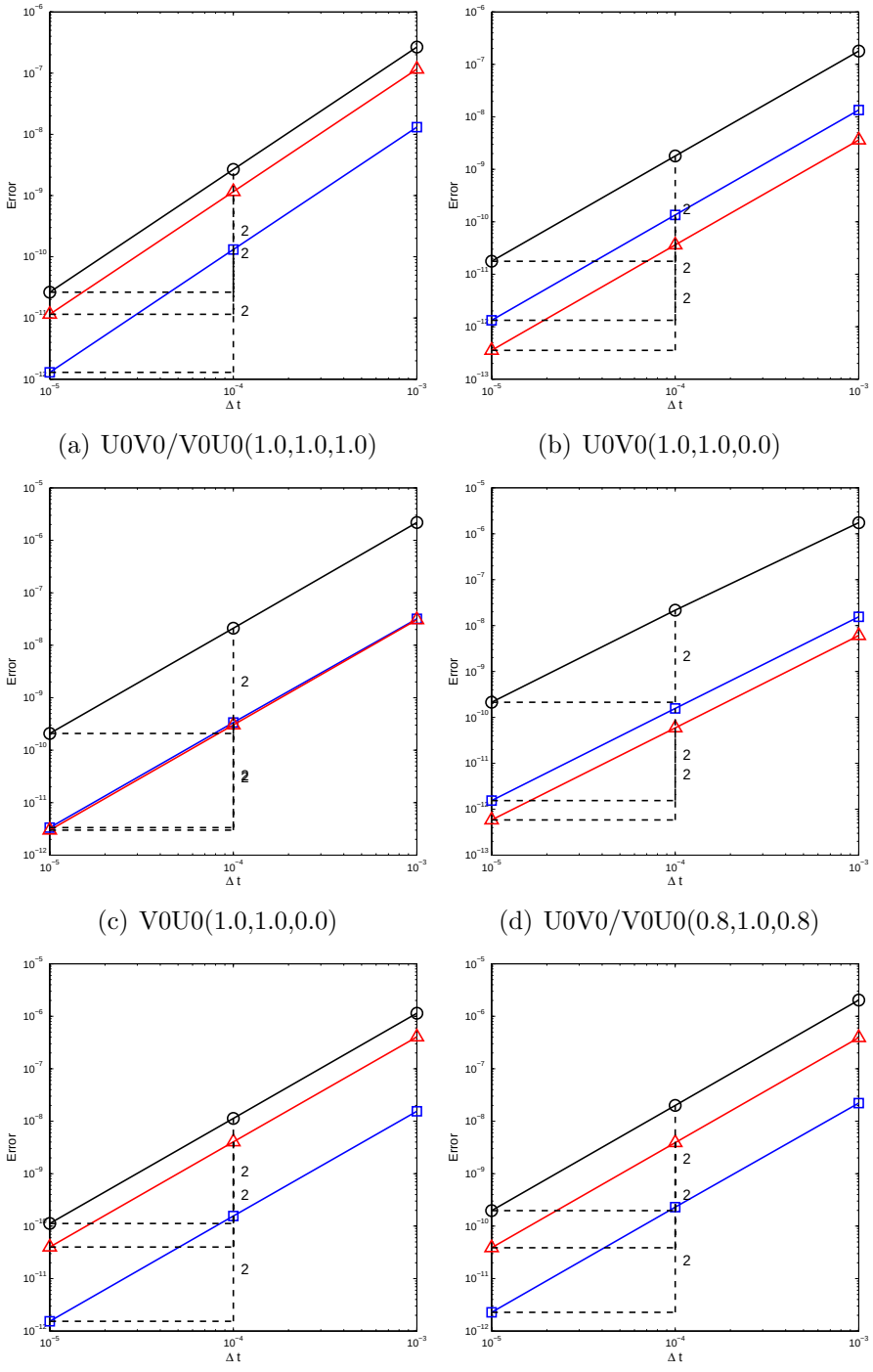
(c) V0U0(1.0,1.0,0.0)

(d) U0V0/V0U0(0.8,1.0,0.8)

(e) U0V1(0.6,0.9,0.2)

(f) V0U1(0.6,0.9,0.2)

Figure 8.23: Time accuracies in the configuration (□), velocity (△), and acceleration (○) in the forced mechani [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 0$ (Option II)]



(a) U0V0/V0U0(1.0,1.0,1.0)

(b) U0V0(1.0,1.0,0.0)

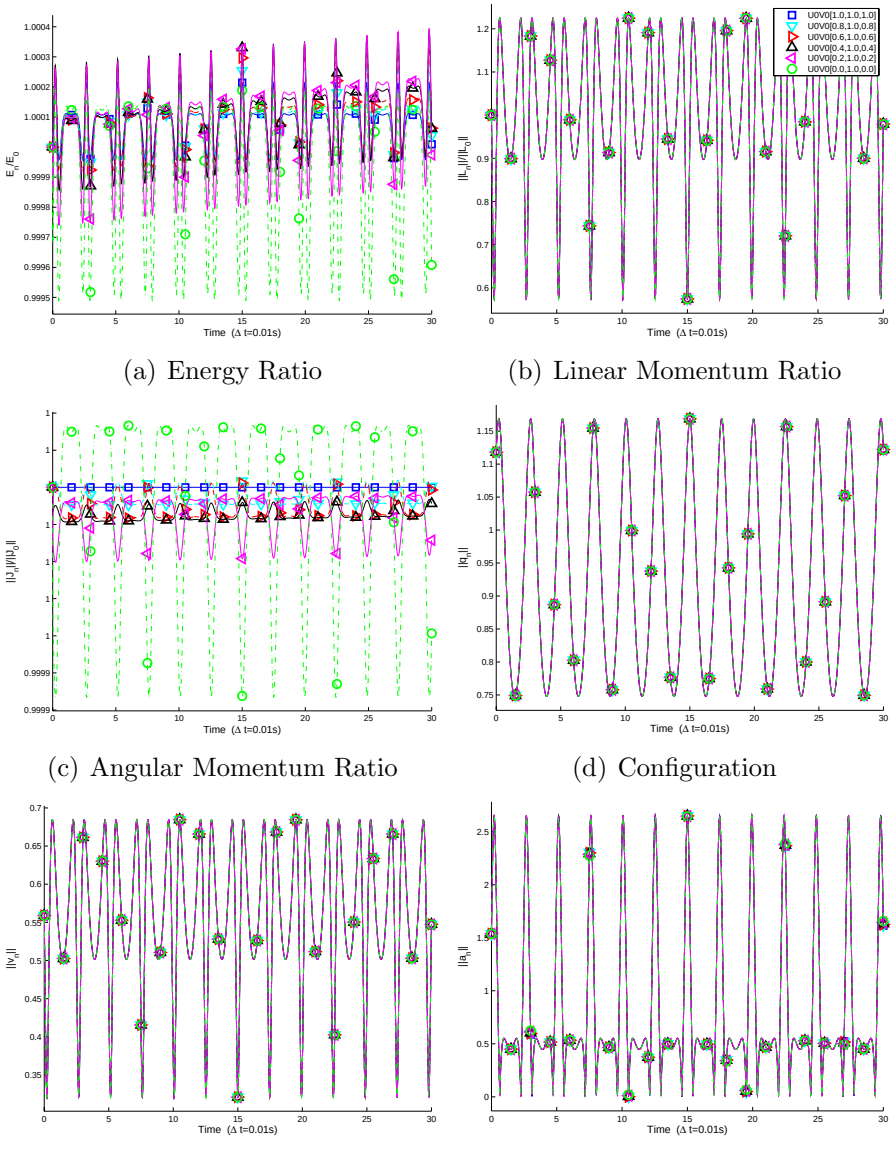
(c) V0U0(1.0,1.0,0.0)

(d) U0V0/V0U0(0.8,1.0,0.8)

(e) U0V1(0.6,0.9,0.2)

(f) V0U1(0.6,0.9,0.2)

Figure 8.24: Time accuracies in the configuration, velocity, and acceleration in the forced mechanical system. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 0$ (Option III)]

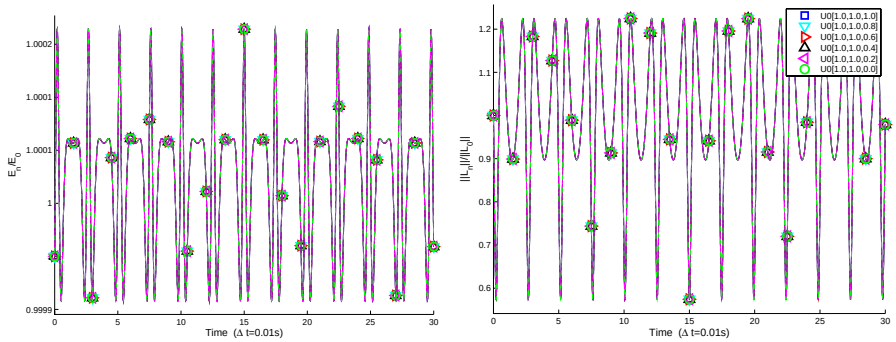


(a) Energy Ratio (b) Linear Momentum Ratio

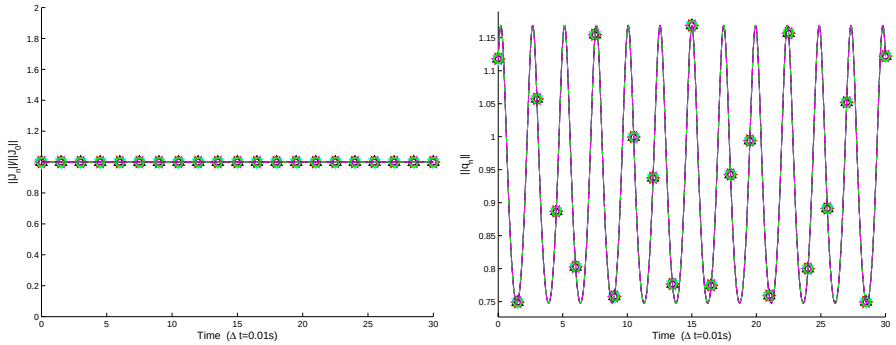
(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

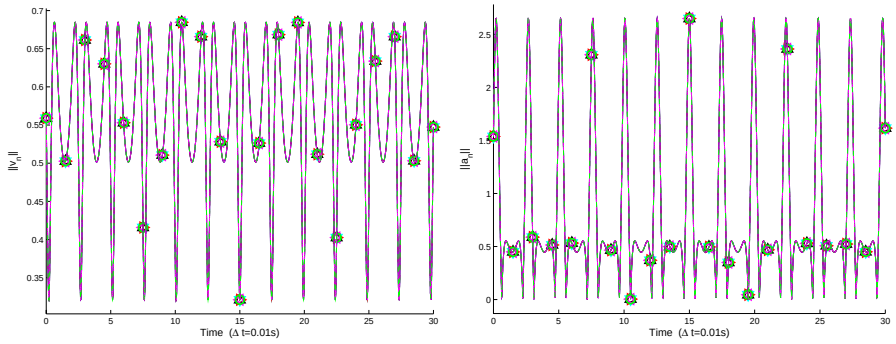
Figure 8.25: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option I) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum Ratio

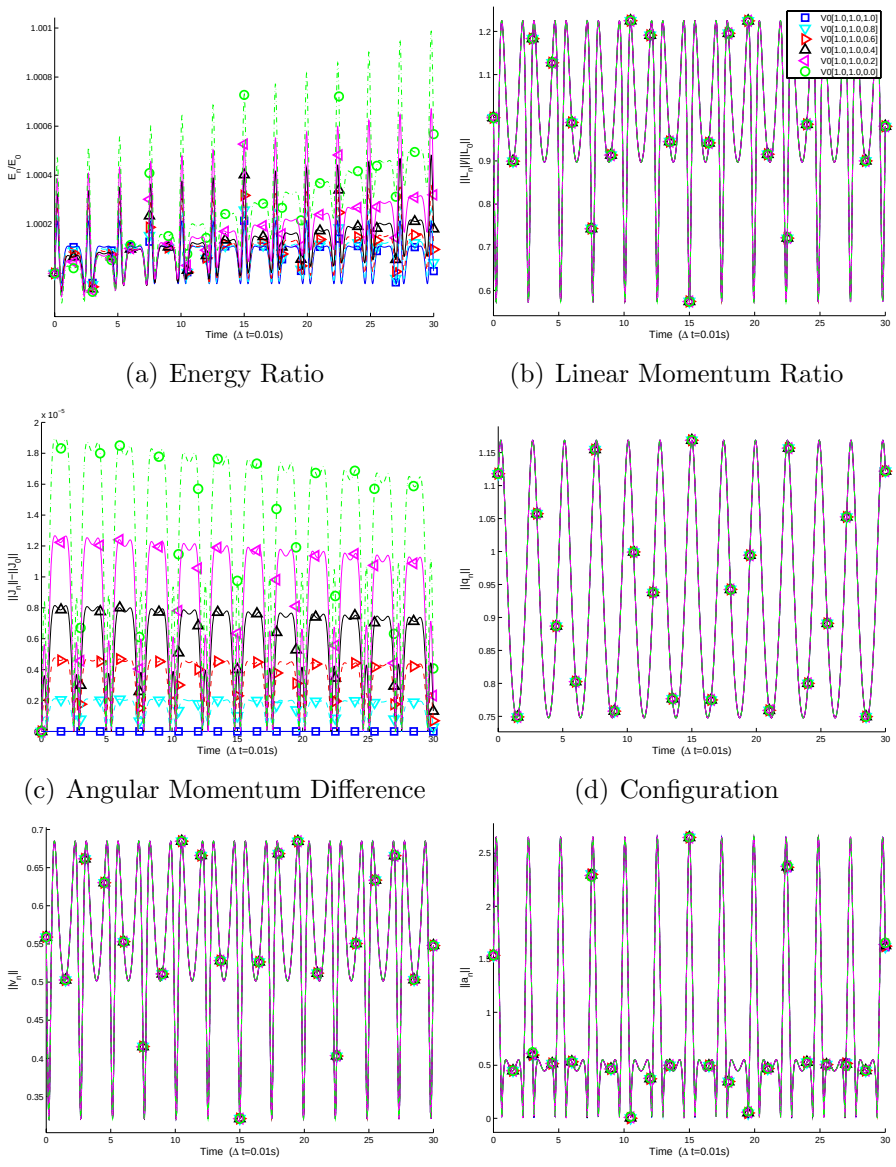


(c) Angular Momentum Ratio (d) Configuration



(e) Velocity Norm (f) Acceleration Norm

Figure 8.26: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option I) - U0V0(1.0,1.0,ρ_∞)]

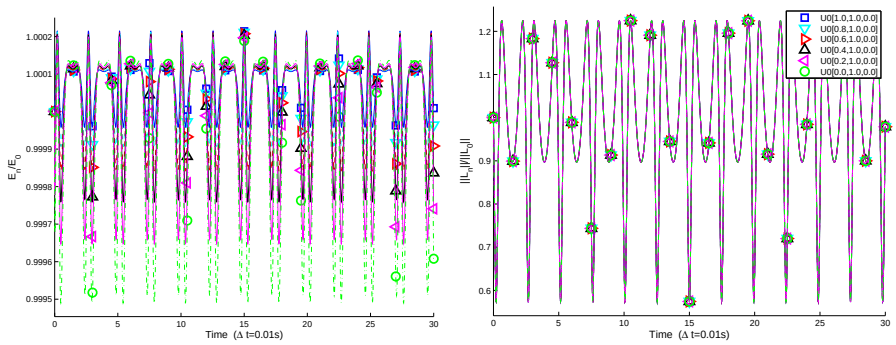


(a) Energy Ratio (b) Linear Momentum Ratio

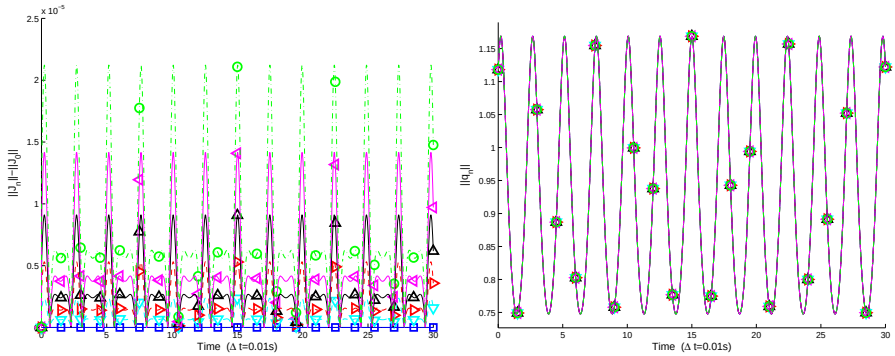
(c) Angular Momentum Difference (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

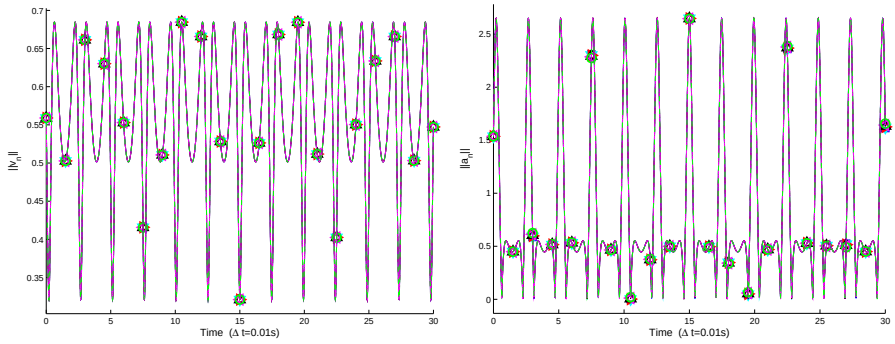
Figure 8.27: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option I) - V0U0(1.0,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum Ratio

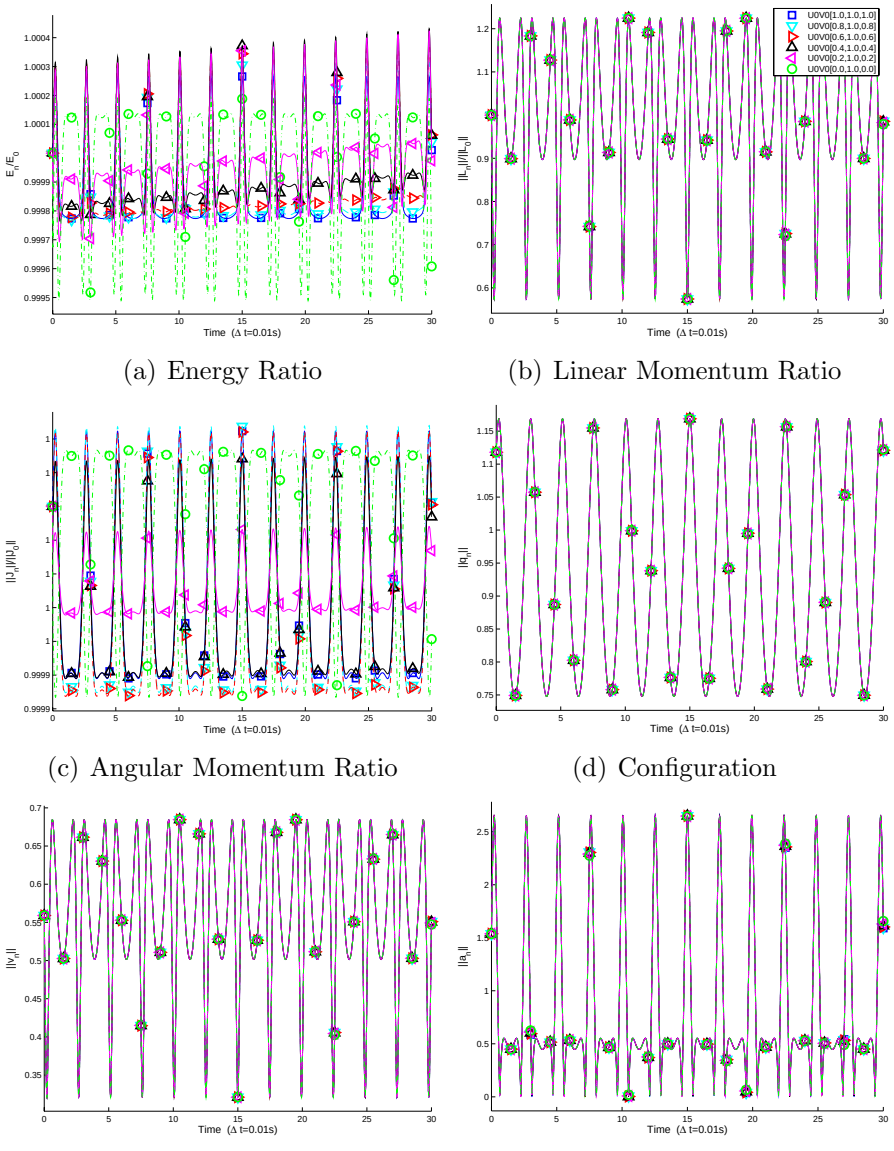


(c) Angular Momentum Difference (d) Configuration



(e) Velocity Norm (f) Acceleration Norm

Figure 8.28: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option I) - U0(ρ_∞,1.0,0.0)]

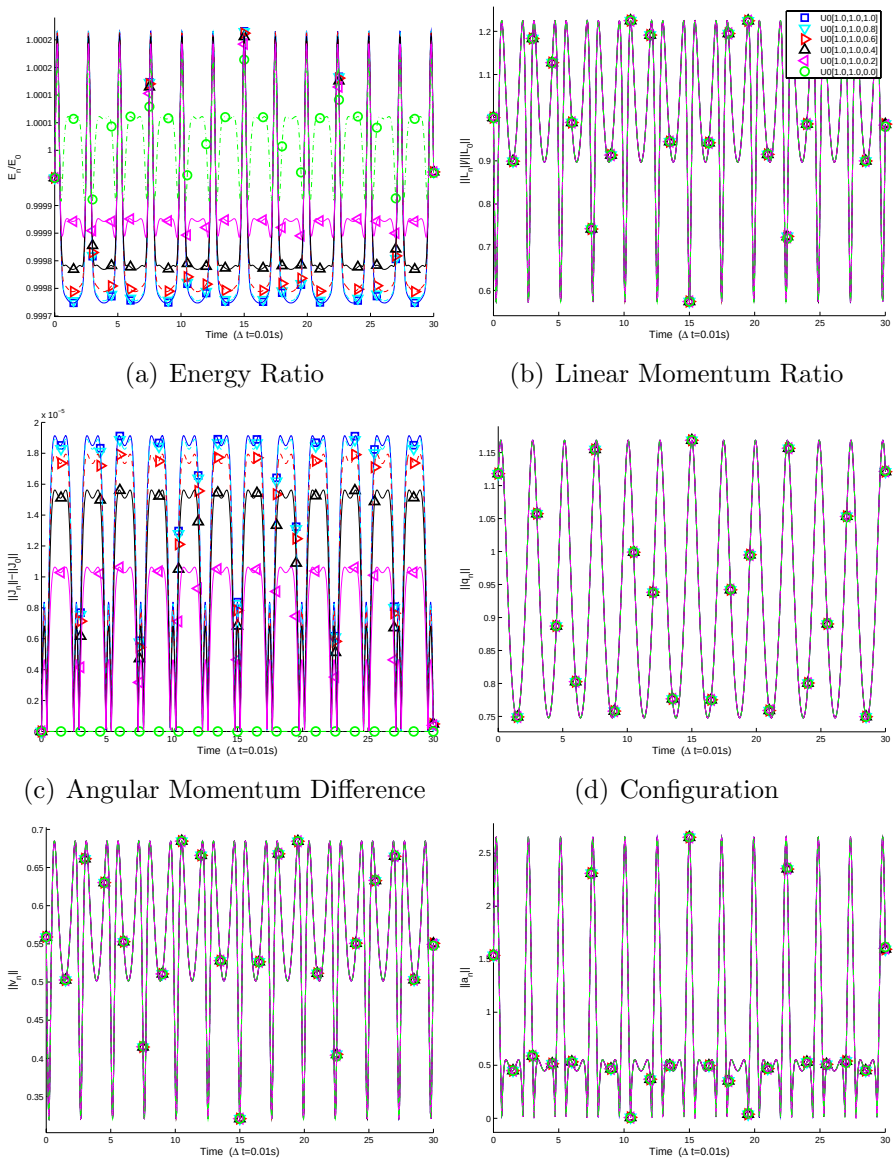


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.29: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option II) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]

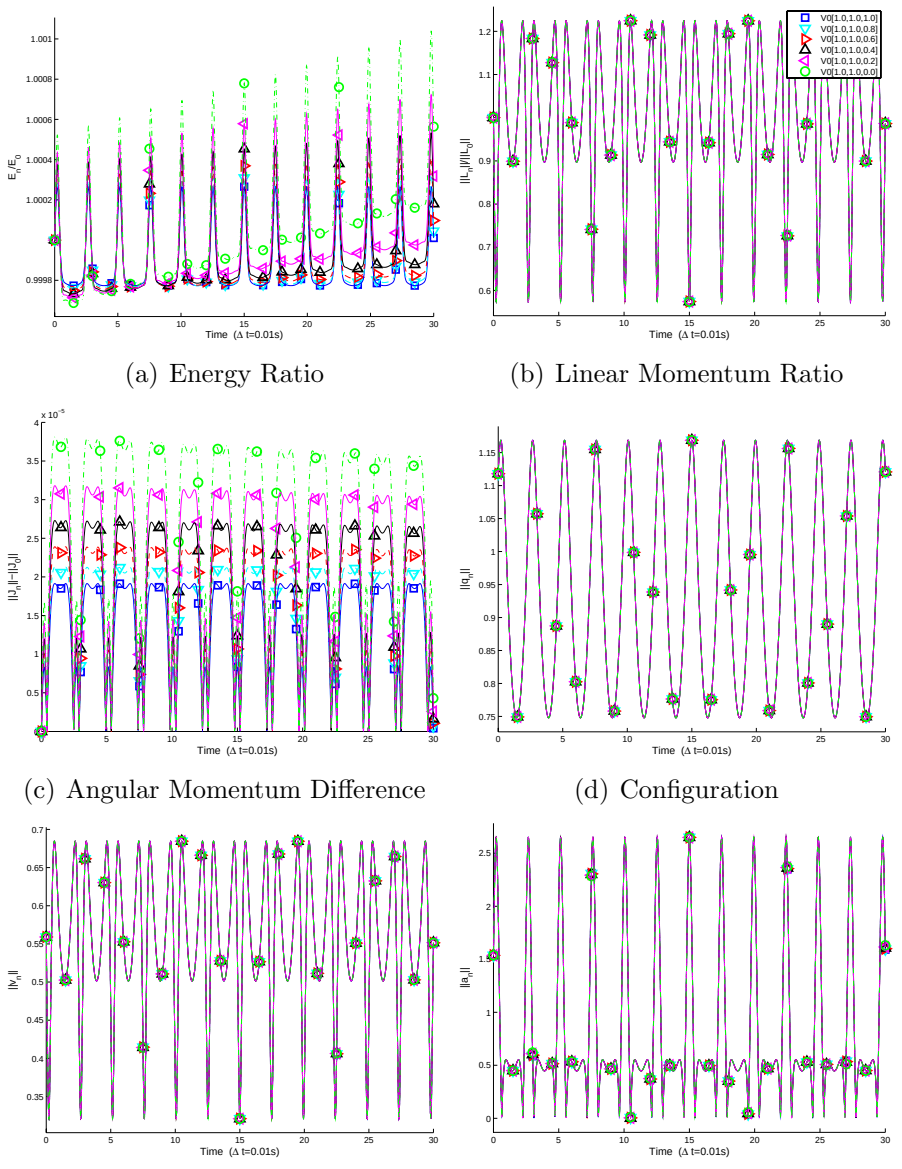


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Difference (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.30: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option II) - U0V0(1.0,1.0,ρ_∞)]

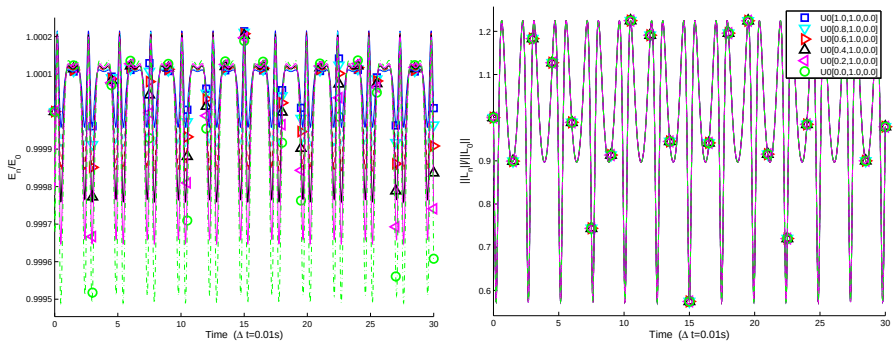


(a) Energy Ratio (b) Linear Momentum Ratio

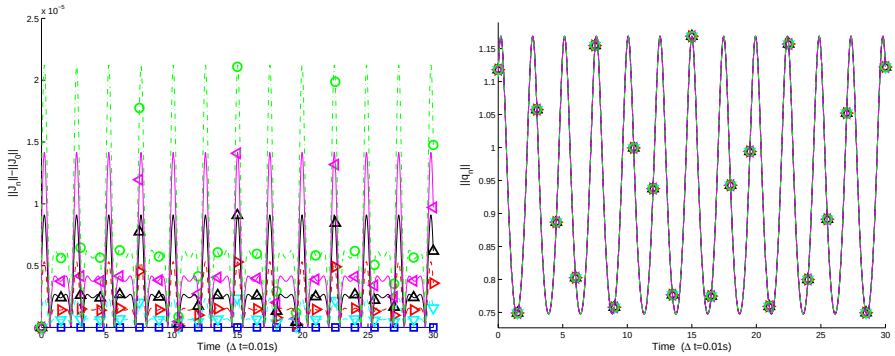
(c) Angular Momentum Difference (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

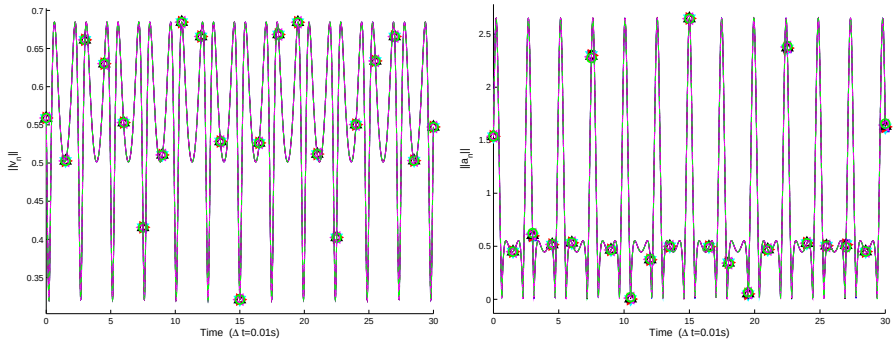
Figure 8.31: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option II) - V0U0(1.0,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum Ratio

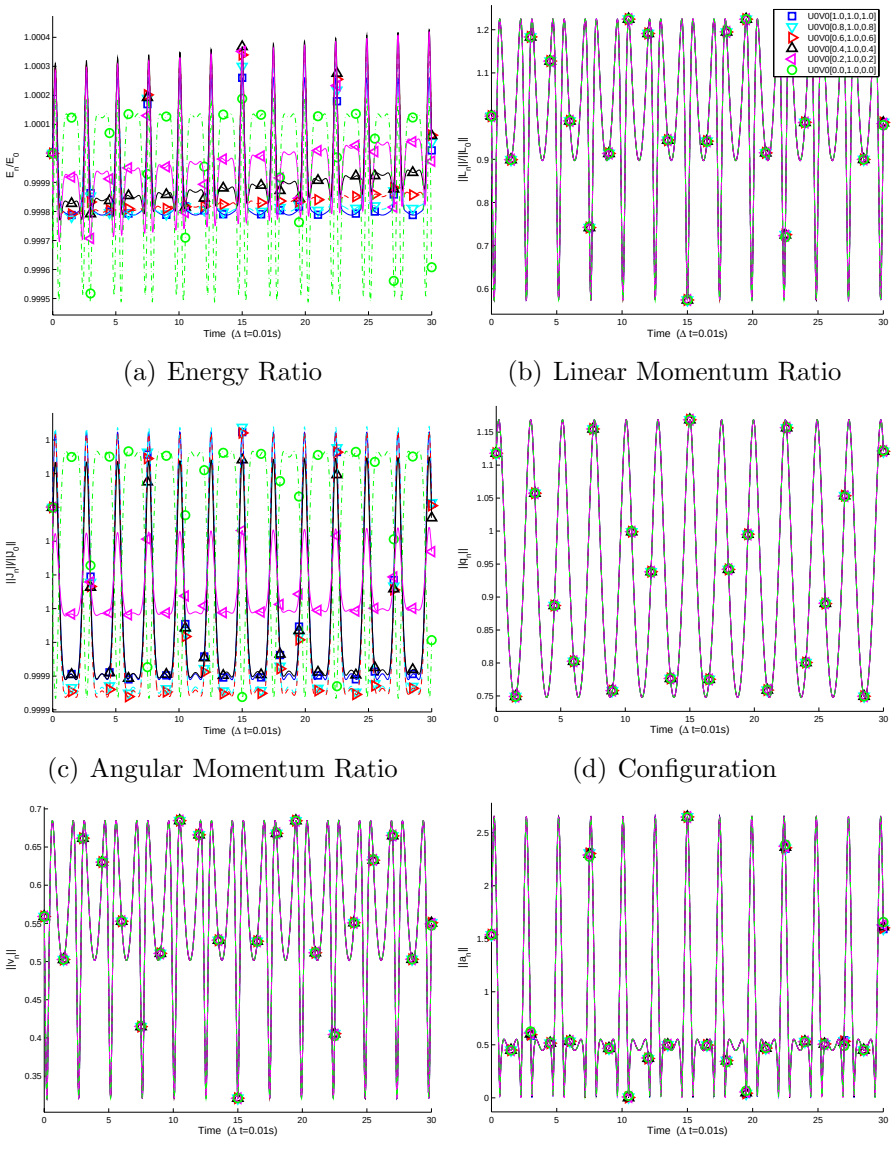


(c) Angular Momentum Difference (d) Configuration



(e) Velocity Norm (f) Acceleration Norm

Figure 8.32: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option II) - U0(ρ_∞,1.0,0.0)]

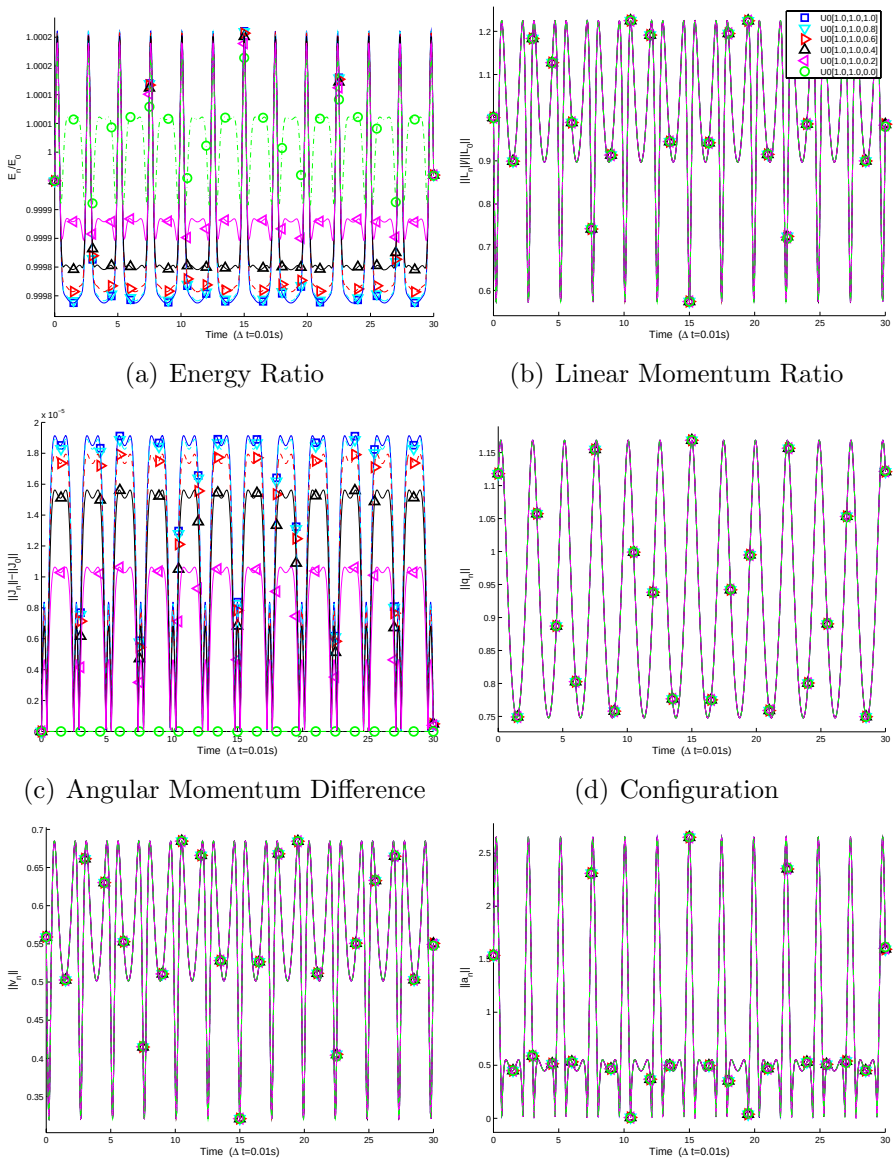


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.33: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option III) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]

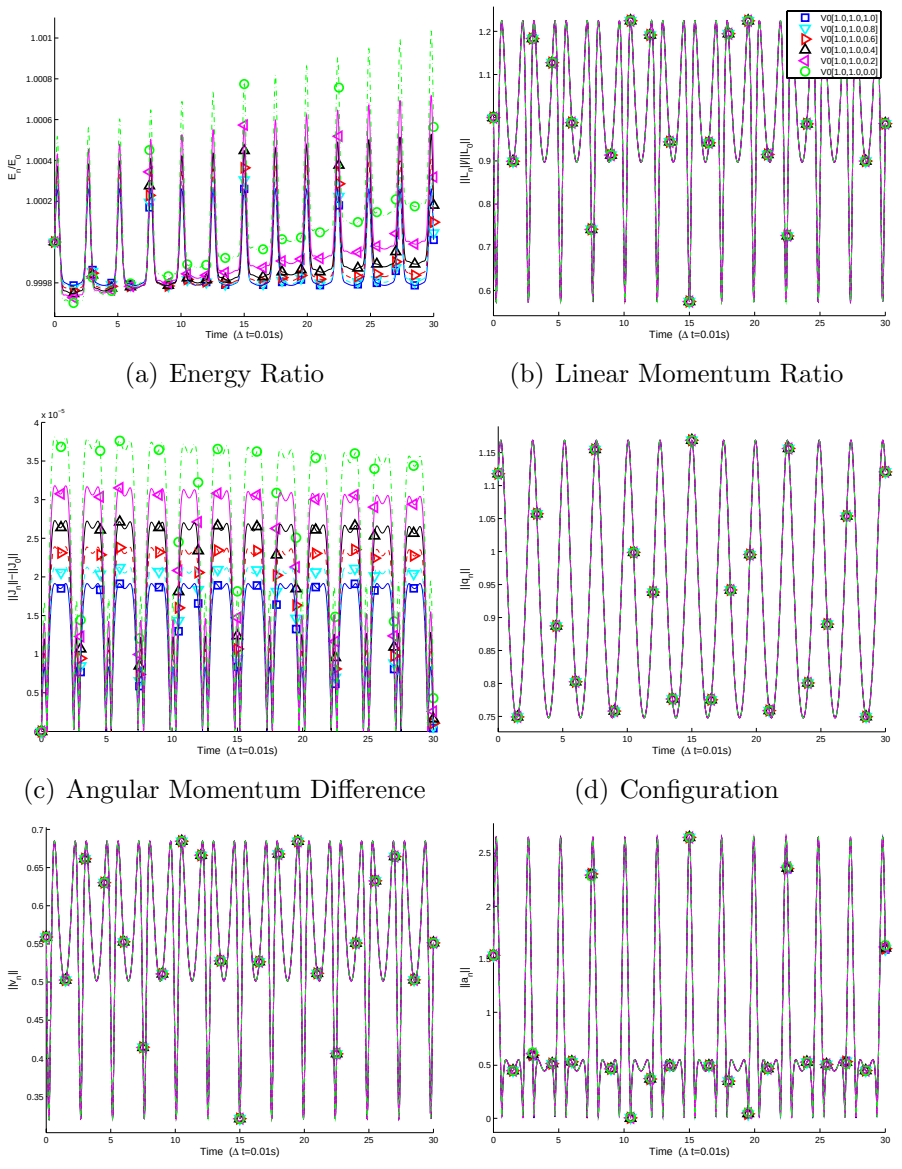


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Difference (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.34: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option III) - U0V0(1.0,1.0,ρ_∞)]

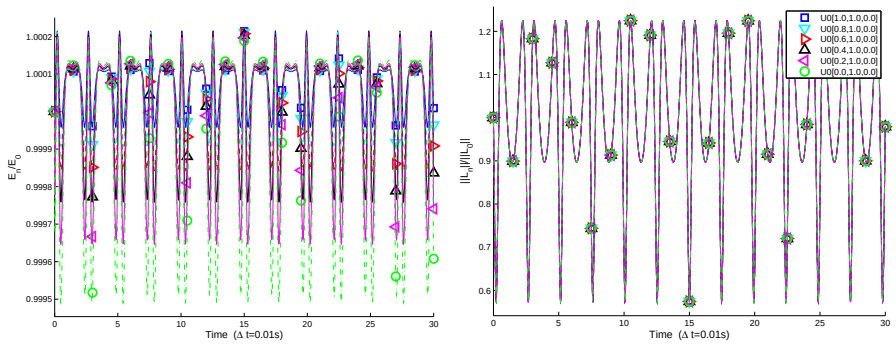


(a) Energy Ratio (b) Linear Momentum Ratio

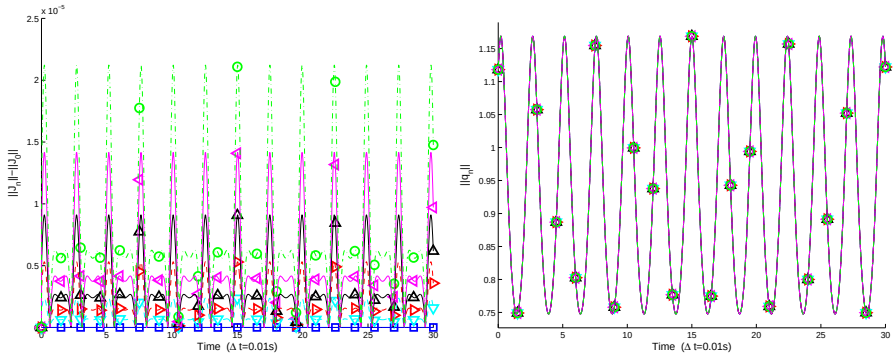
(c) Angular Momentum Difference (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

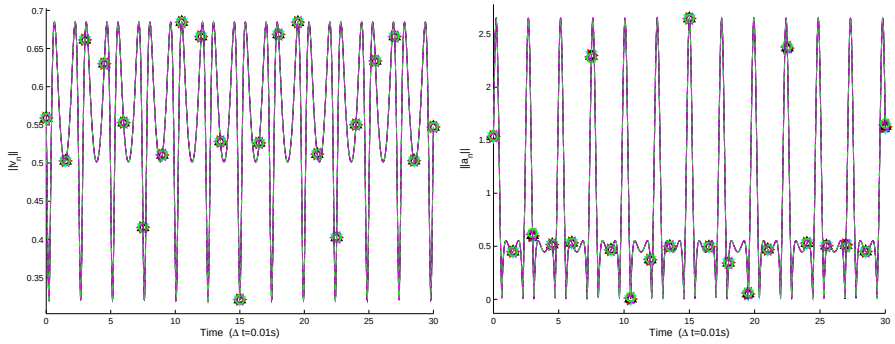
Figure 8.35: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option III) - V0U0(1.0,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum Ratio

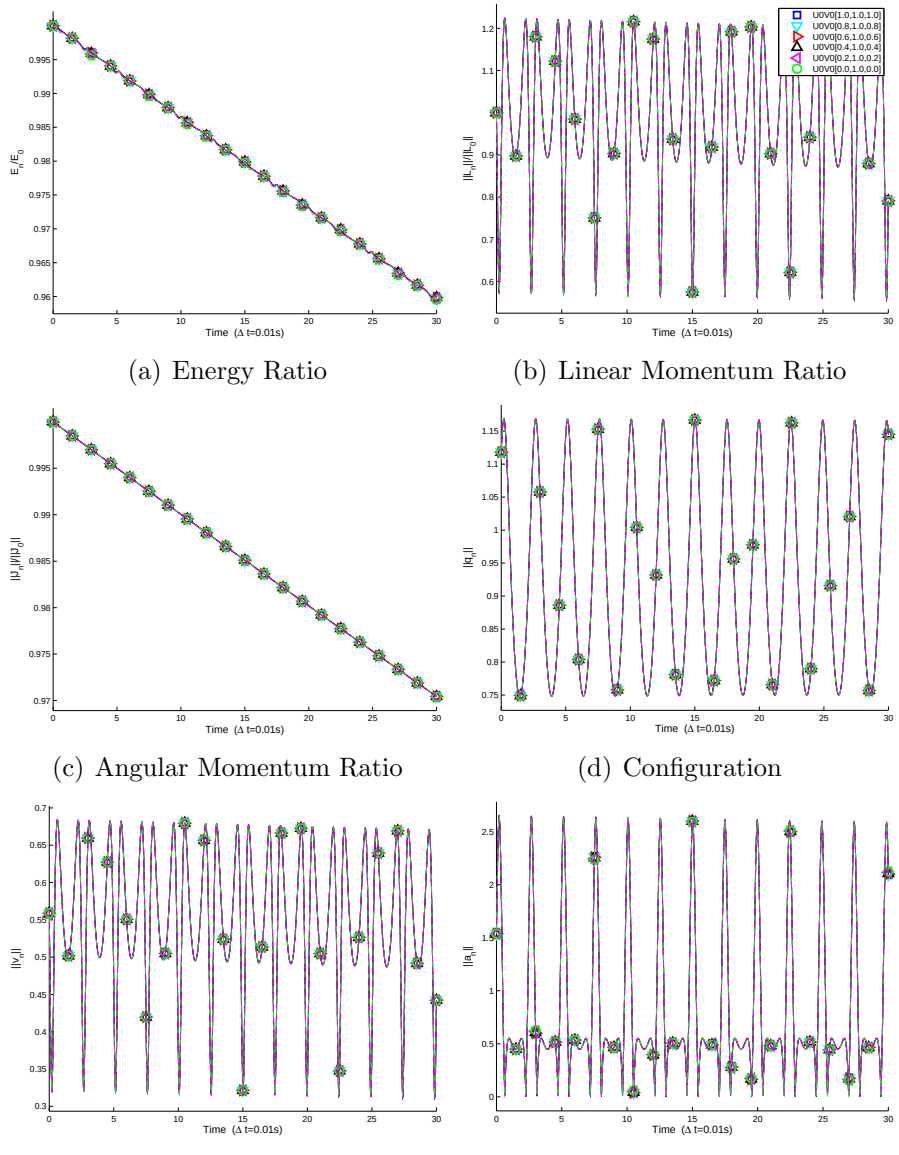


(c) Angular Momentum Difference (d) Configuration



(e) Velocity Norm (f) Acceleration Norm

Figure 8.36: Time histories in the ***conservative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option III) - U0(ρ_∞,1.0,0.0)]



(a) Energy Ratio

(b) Linear Momentum Ratio

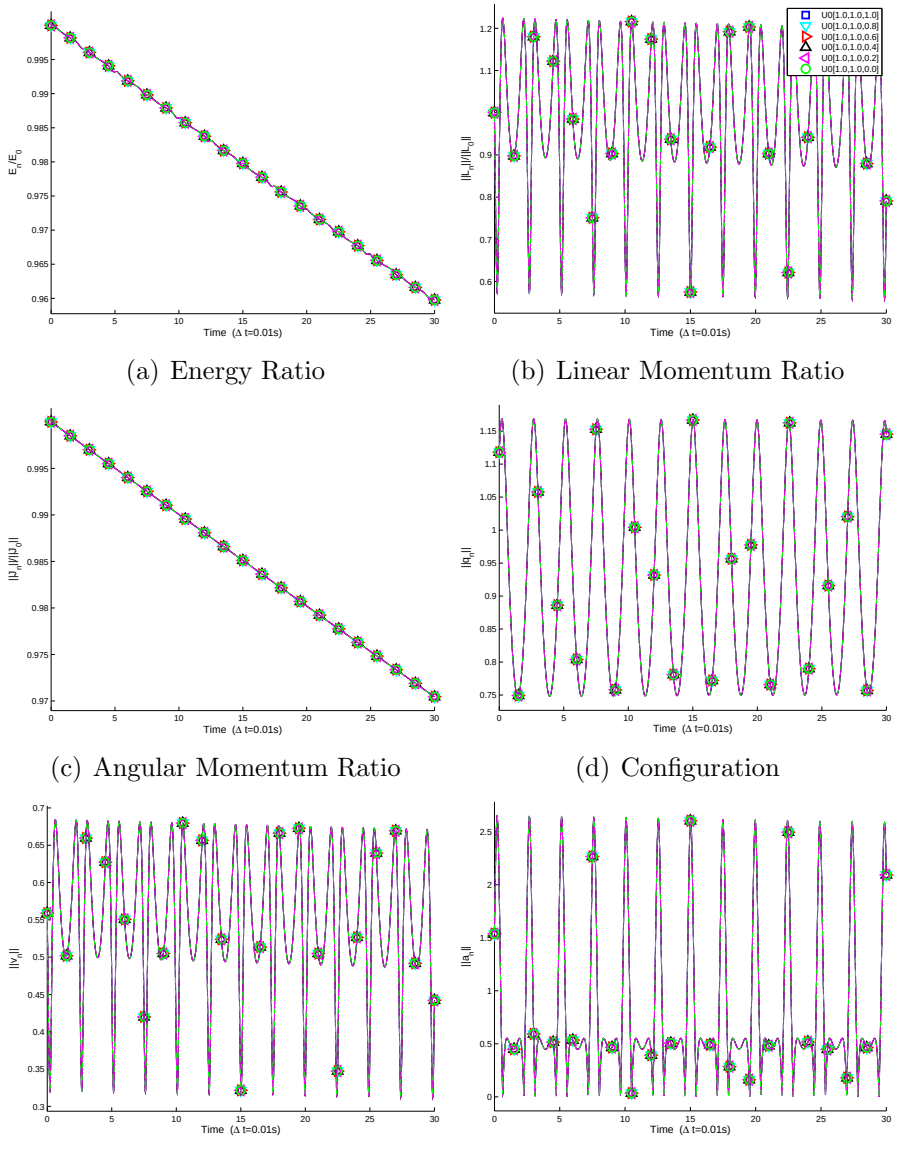
(c) Angular Momentum Ratio

(d) Configuration

(e) Velocity Norm

(f) Acceleration Norm

Figure 8.37: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 0$ (Option I) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]

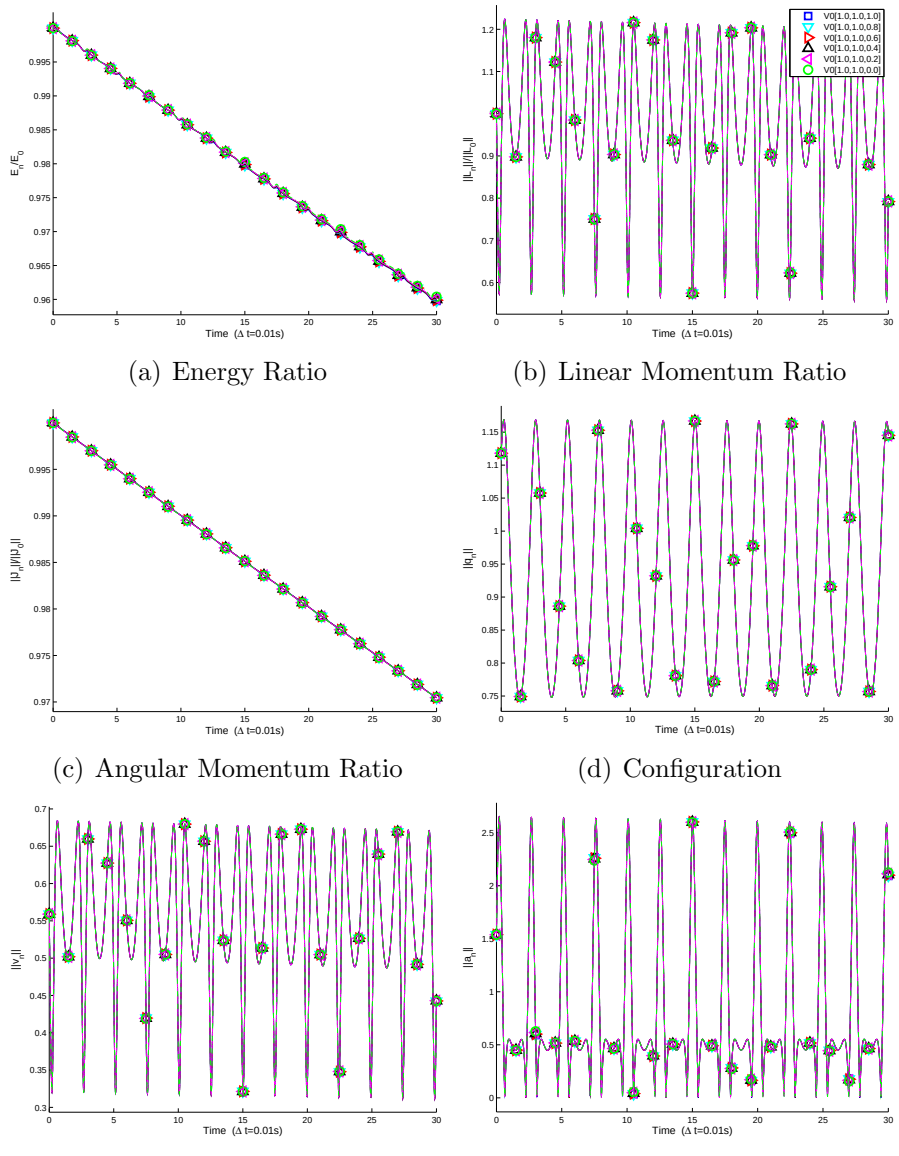


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.38: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 0$ (Option I) - U0V0(1.0,1.0,ρ_∞)]

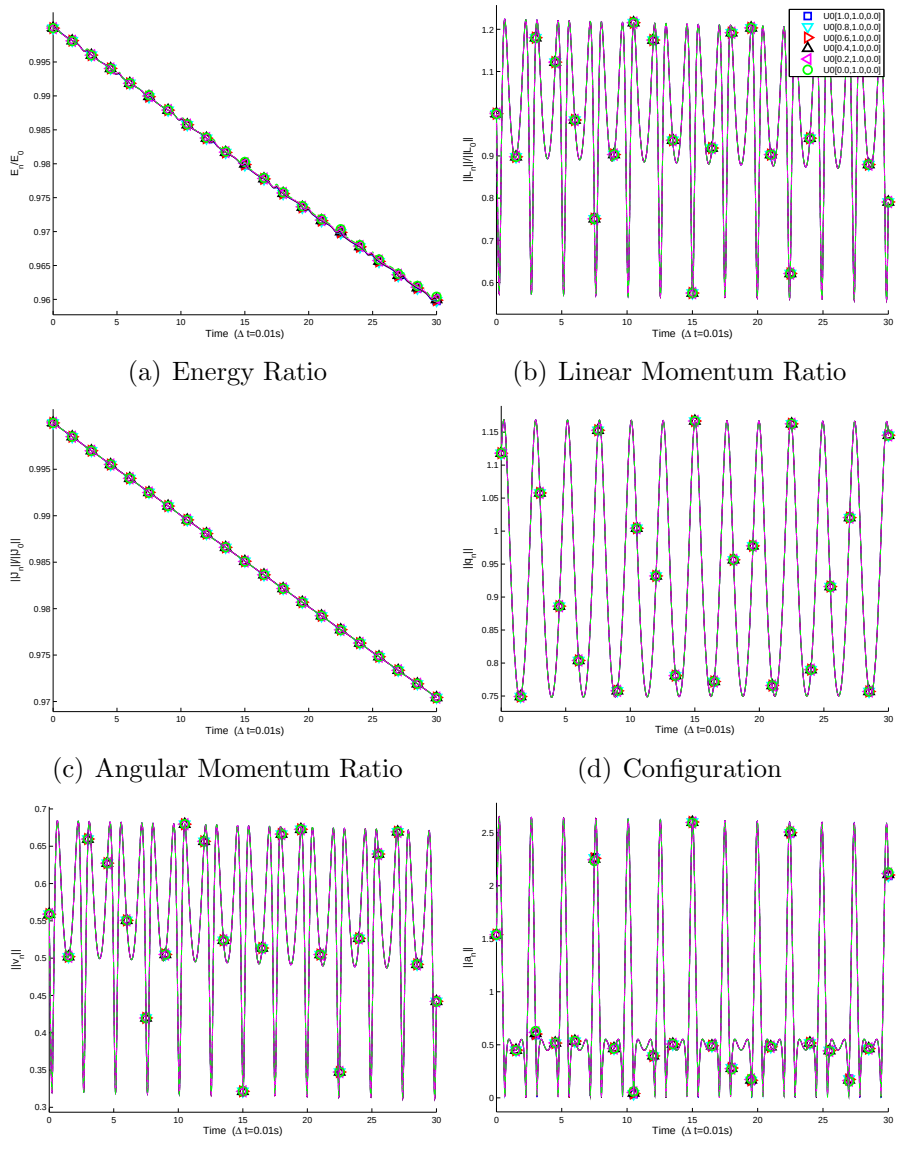


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.39: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 0$ (Option I) - V0U0(1.0,1.0,ρ_∞)]

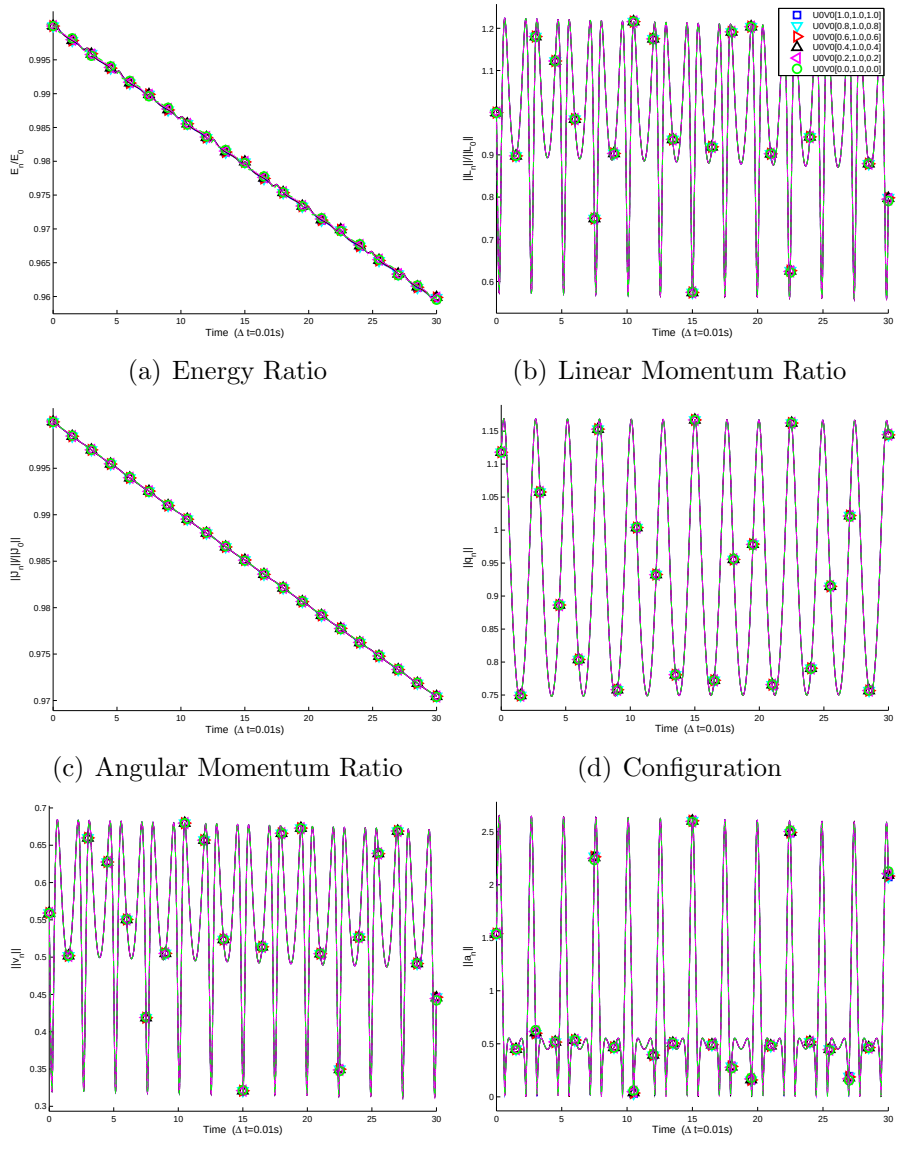


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.40: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 0$ (Option I) - U0(ρ_∞,1.0,0.0)]



(a) Energy Ratio

(b) Linear Momentum Ratio

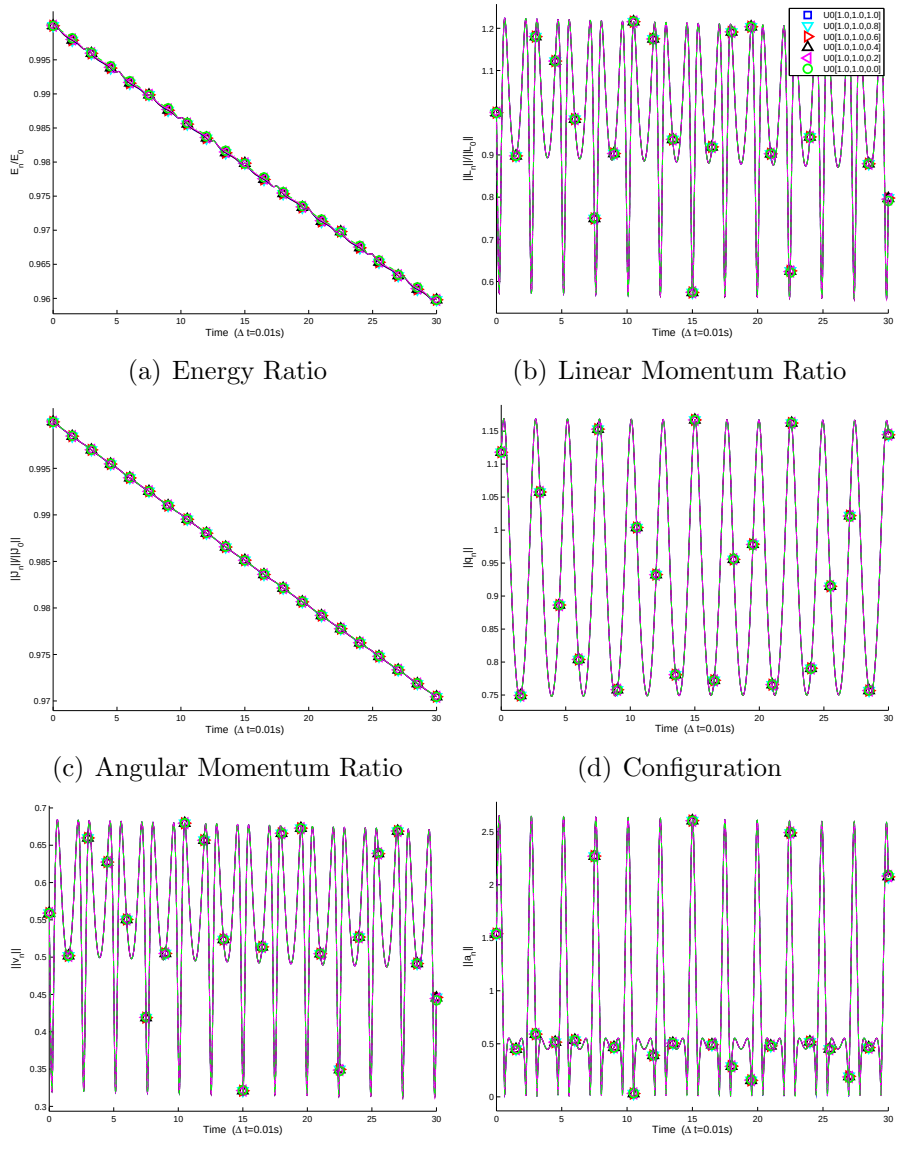
(c) Angular Momentum Ratio

(d) Configuration

(e) Velocity Norm

(f) Acceleration Norm

Figure 8.41: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 0$ (Option II) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]

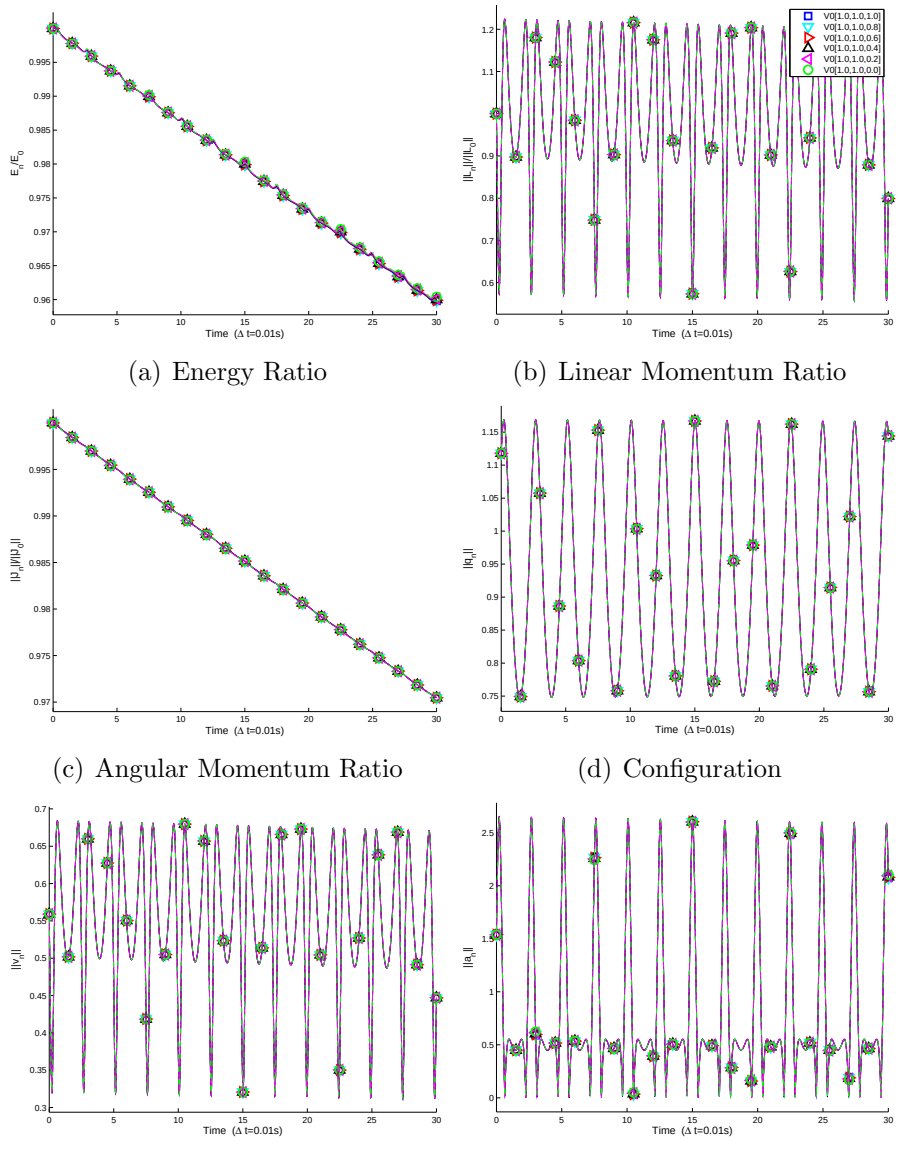


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.42: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 0$ (Option II) - U0V0(1.0,1.0,ρ_∞)]

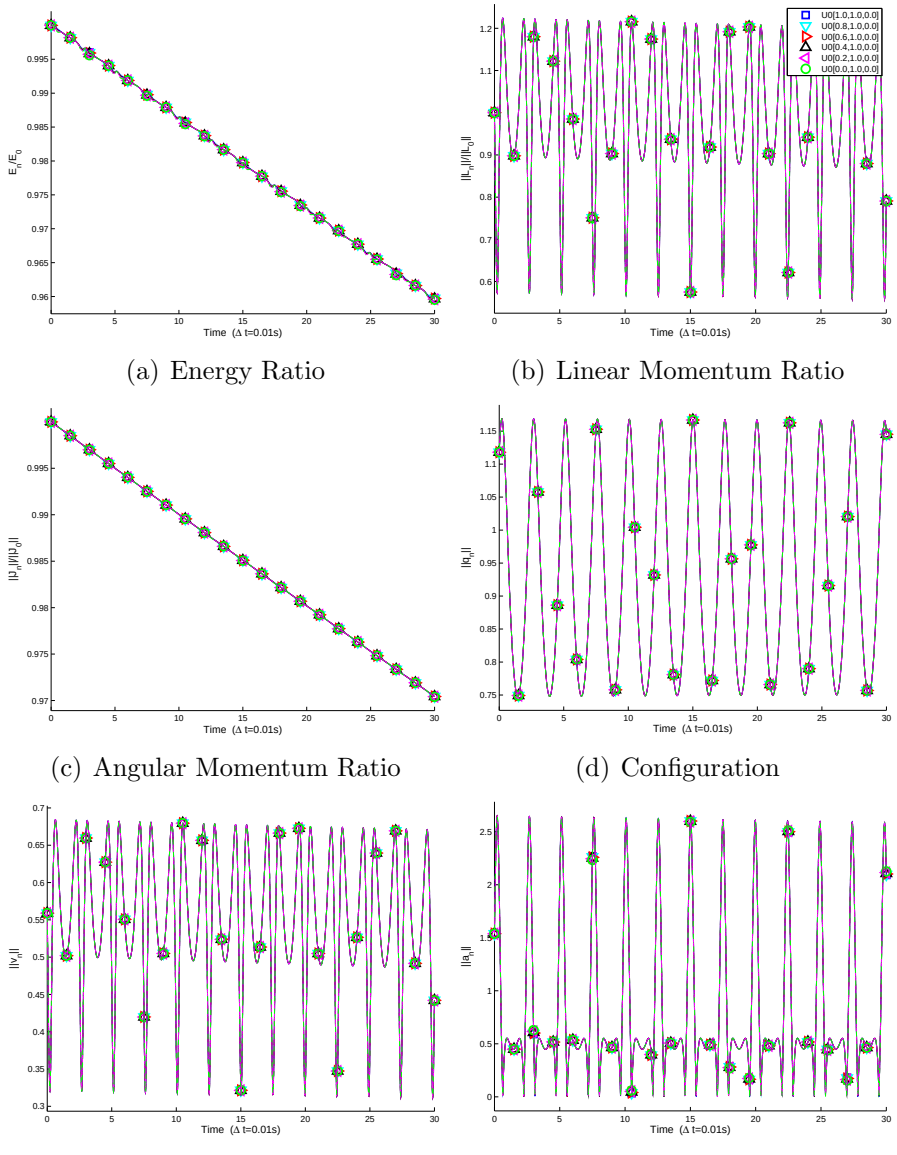


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.43: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 0$ (Option II) - V0U0(1.0,1.0,ρ_∞)]

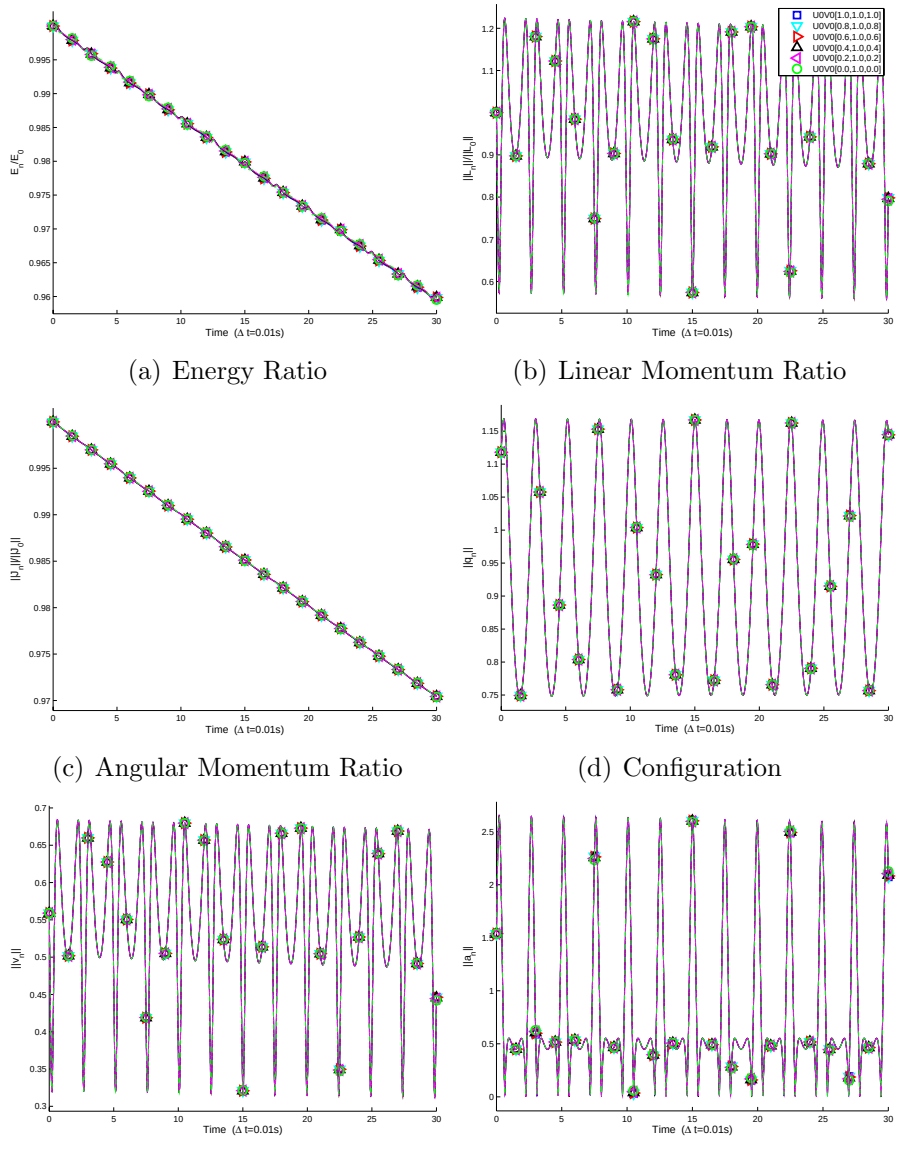


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.44: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 0$ (Option II) - U0(ρ_∞,1.0,0.0)]

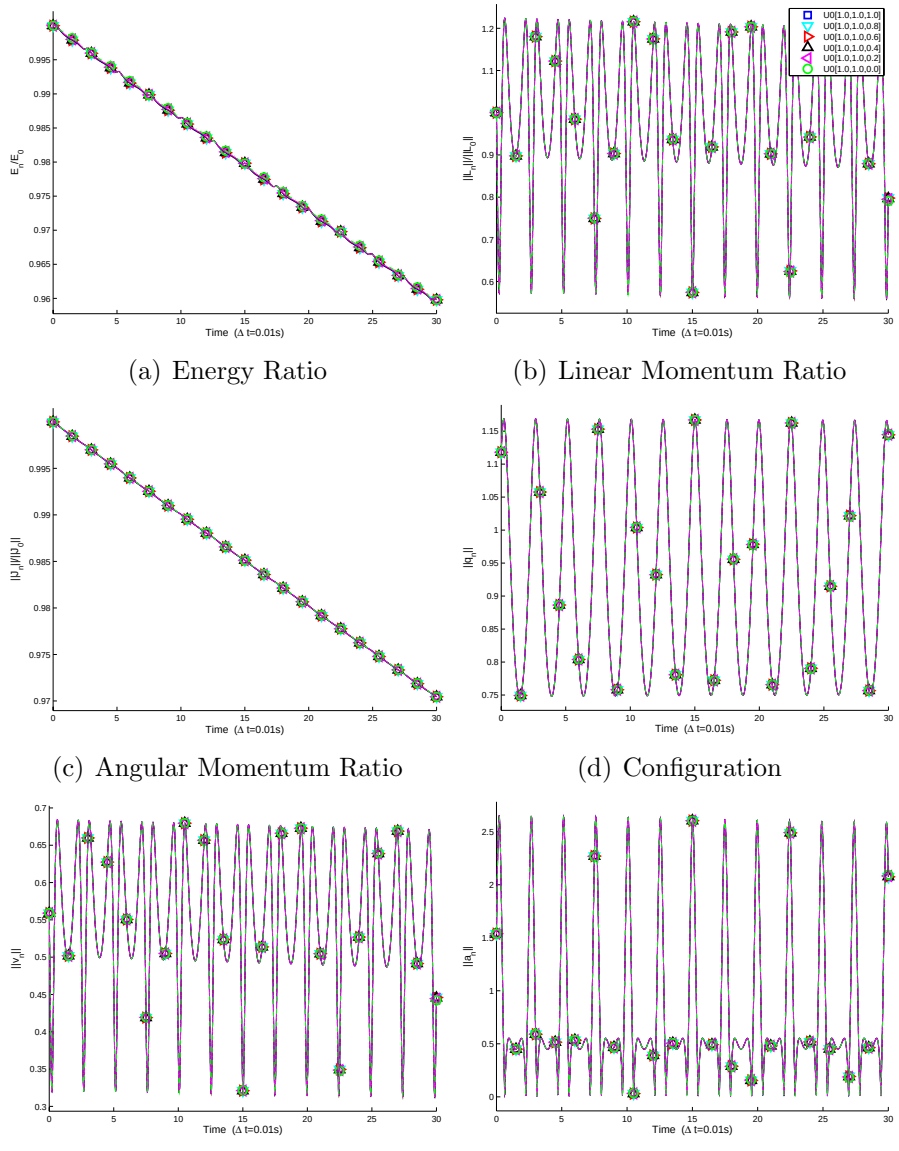


(a) Energy Ratio (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.45: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 0$ (Option III) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Energy Ratio

(b) Linear Momentum Ratio

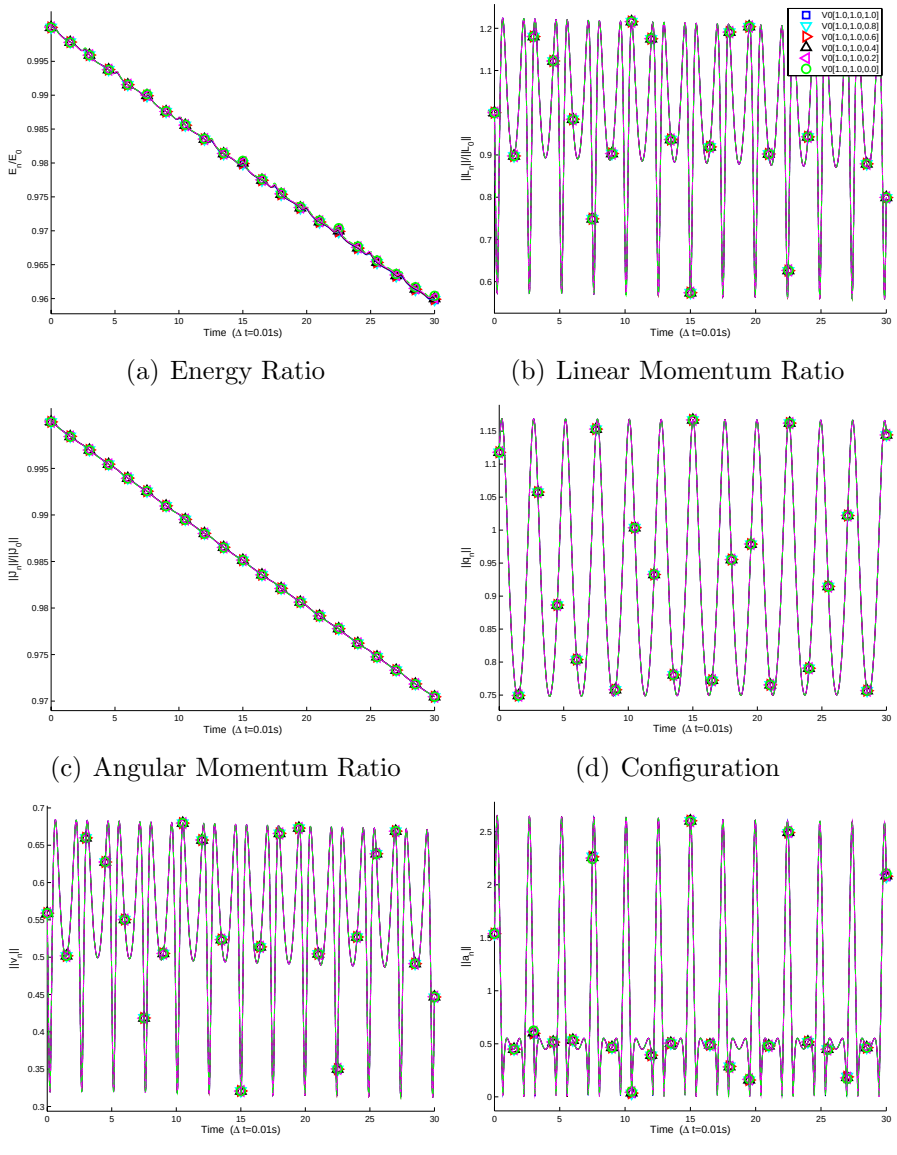
(c) Angular Momentum Ratio

(d) Configuration

(e) Velocity Norm

(f) Acceleration Norm

Figure 8.46: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 0$ (Option III) - U0V0(1.0,1.0,ρ_∞)]



(a) Energy Ratio

(b) Linear Momentum Ratio

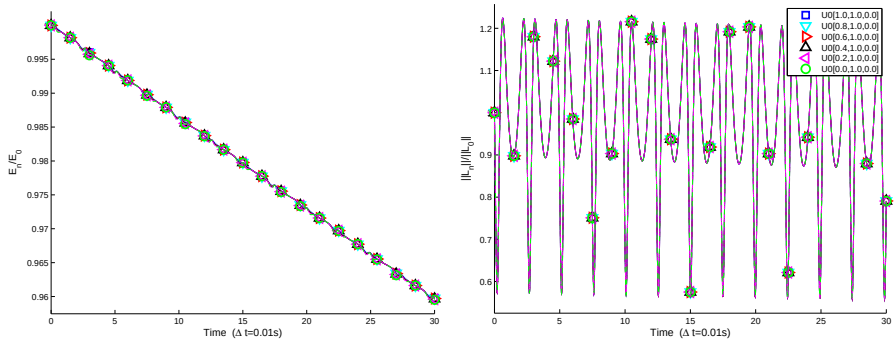
(c) Angular Momentum Ratio

(d) Configuration

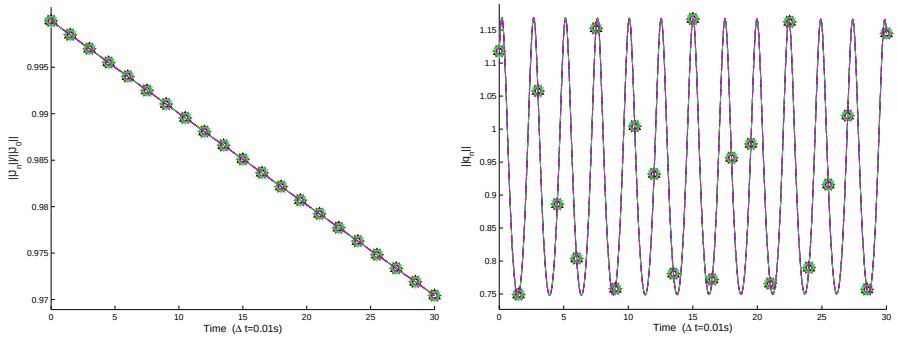
(e) Velocity Norm

(f) Acceleration Norm

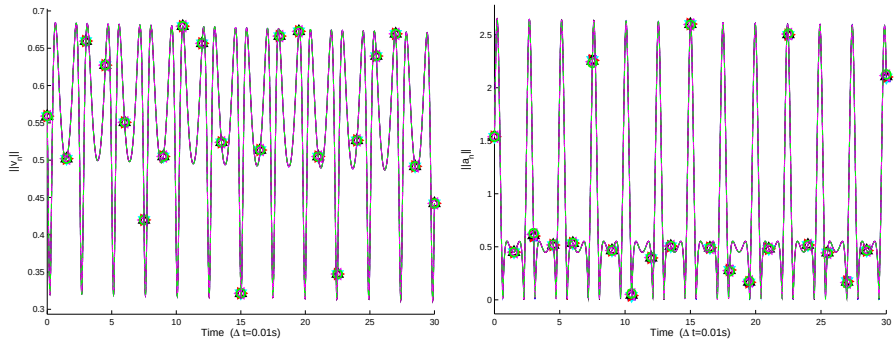
Figure 8.47: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 0$ (Option III) - V0U0(1.0,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum Ratio

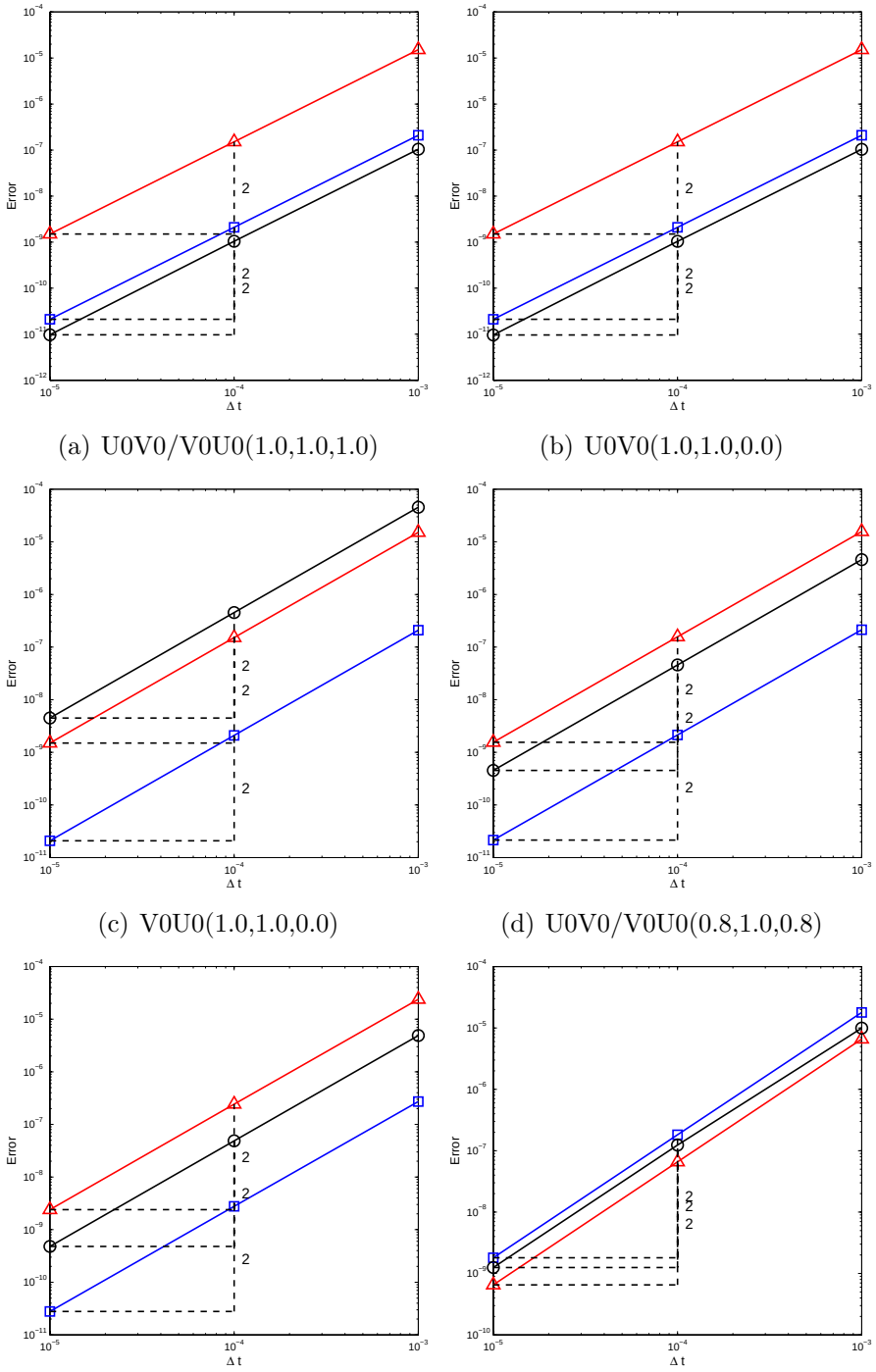


(c) Angular Momentum Ratio (d) Configuration



(e) Velocity Norm (f) Acceleration Norm

Figure 8.48: Time histories in the ***dissipative system***. [Problem: Nonlinear oscillator] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 0$ (Option III) - U0(ρ_∞,1.0,0.0)]



(a) U0V0/V0U0(1.0,1.0,1.0)

(b) U0V0(1.0,1.0,0.0)

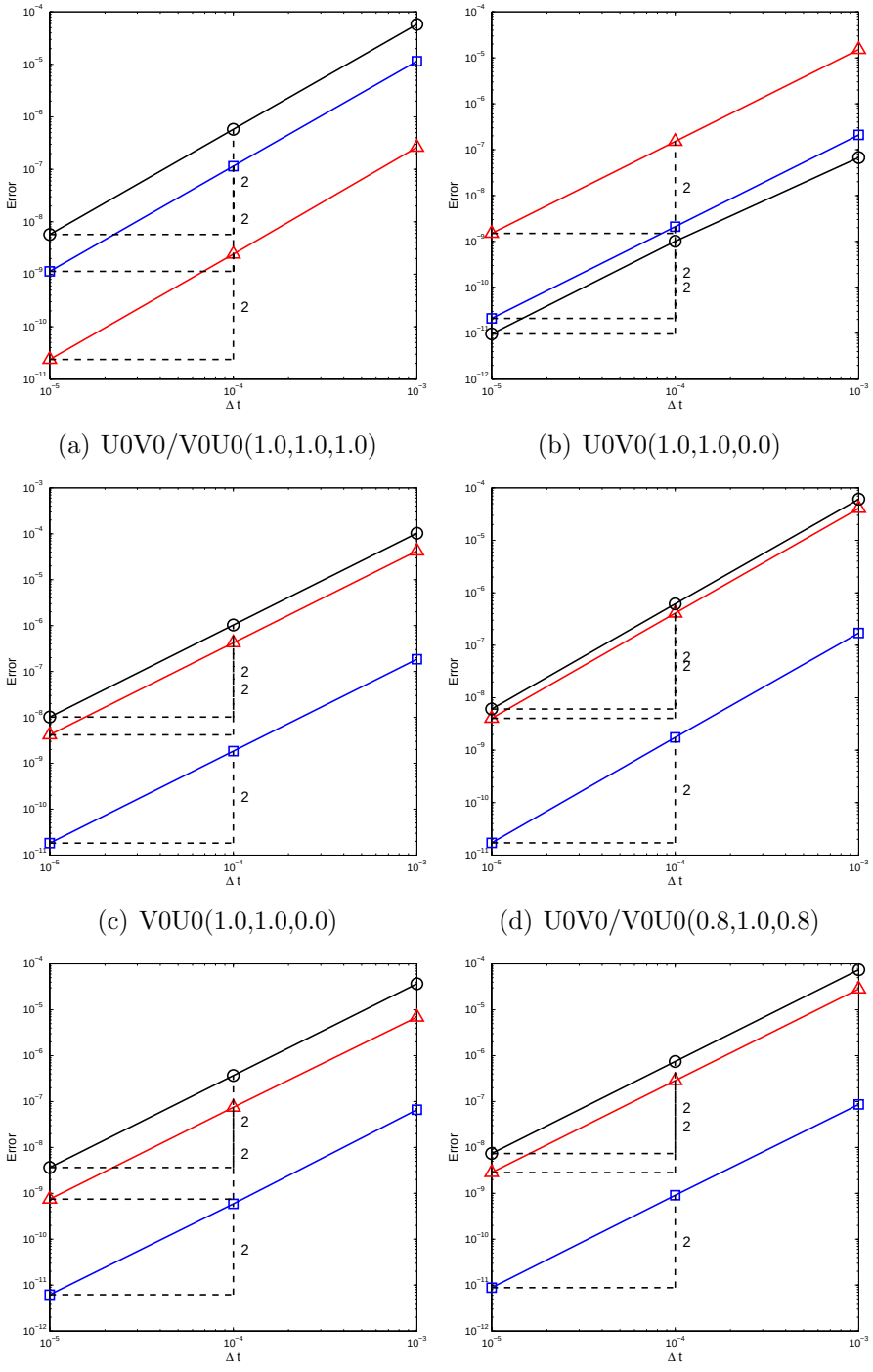
(c) V0U0(1.0,1.0,0.0)

(d) U0V0/V0U0(0.8,1.0,0.8)

(e) U0V1(0.6,0.9,0.2)

(f) V0U1(0.6,0.9,0.2)

Figure 8.49: Time accuracies in the configuration (□), velocity (△), and acceleration (○) in the forced mechanical system. [Problem: Kepler's problem] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 1$ (Option I)]



(a) U0V0/V0U0(1.0,1.0,1.0)

(b) U0V0(1.0,1.0,0.0)

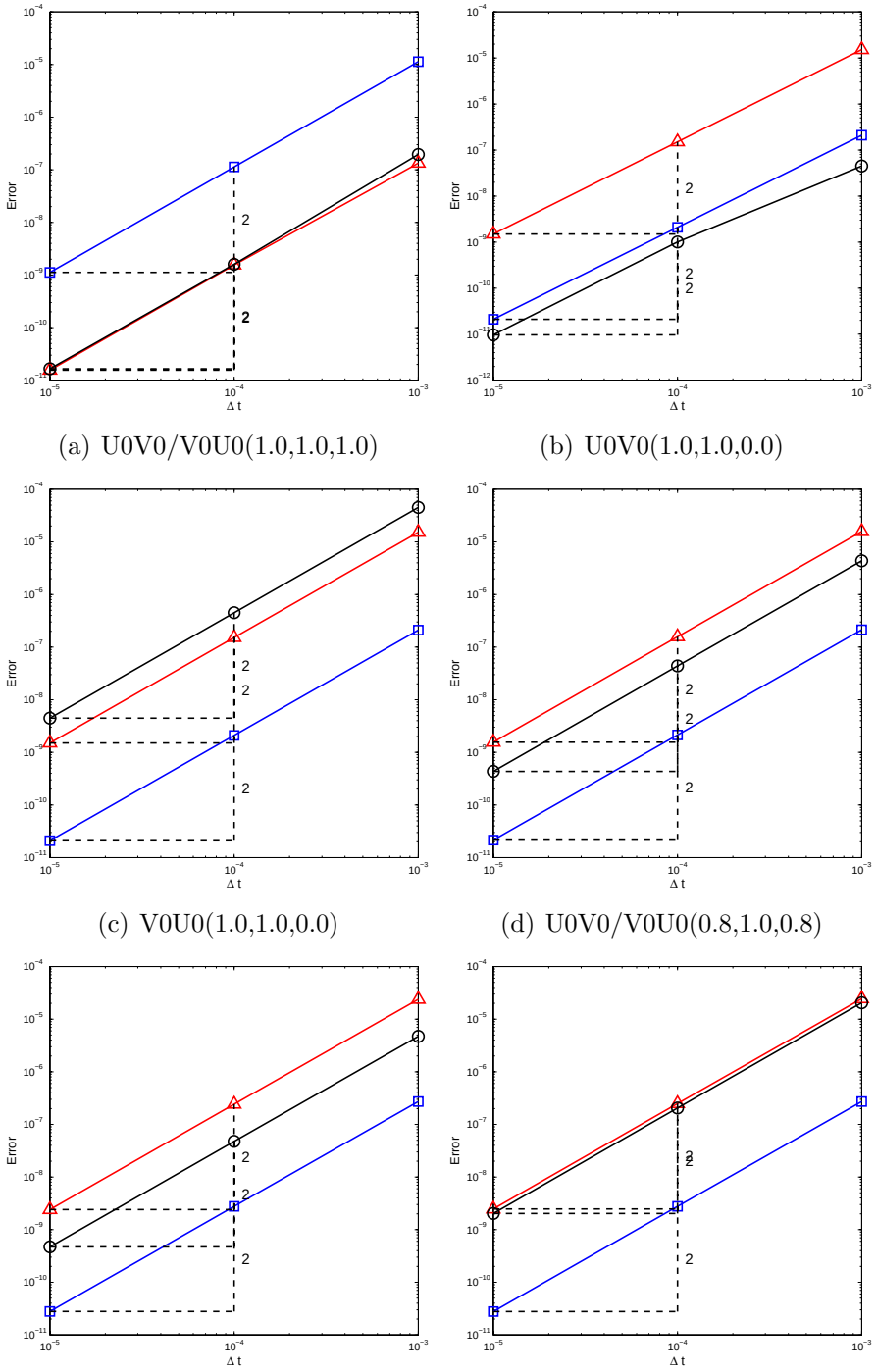
(c) V0U0(1.0,1.0,0.0)

(d) U0V0/V0U0(0.8,1.0,0.8)

(e) U0V1(0.6,0.9,0.2)

(f) V0U1(0.6,0.9,0.2)

Figure 8.50: Time accuracies in the configuration (□), velocity (△), and acceleration (○) in the forced mechani [Problem: Kepler's problem] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 1$ (Option II)]



(a) U0V0/V0U0(1.0,1.0,1.0)

(b) U0V0(1.0,1.0,0.0)

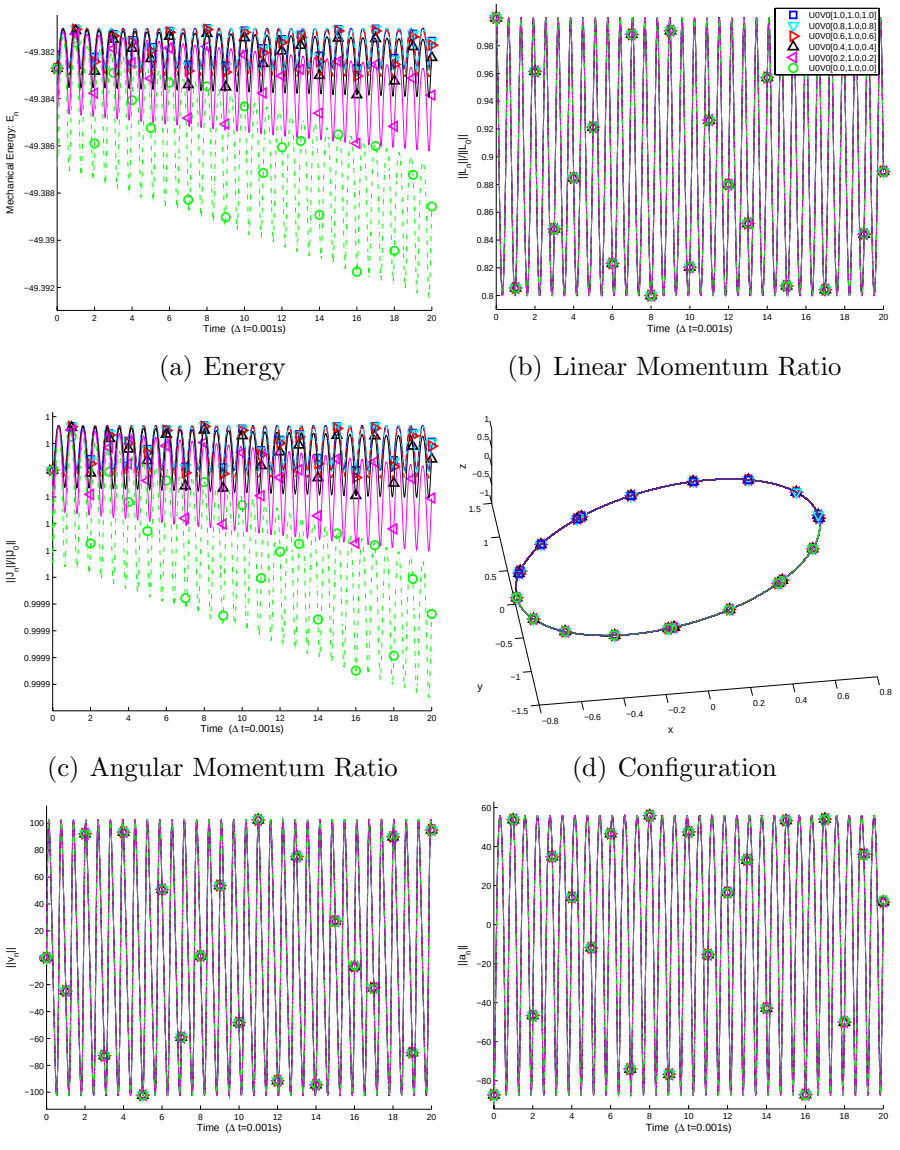
(c) V0U0(1.0,1.0,0.0)

(d) U0V0/V0U0(0.8,1.0,0.8)

(e) U0V1(0.6,0.9,0.2)

(f) V0U1(0.6,0.9,0.2)

Figure 8.51: Time accuracies in the configuration, velocity, and acceleration in the forced mechanical system. [Problem: Kepler's problem] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 1$ (Option III)]

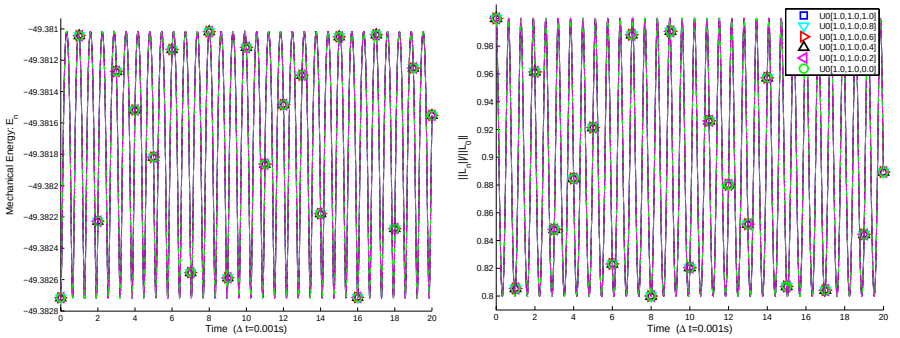


(a) Energy (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

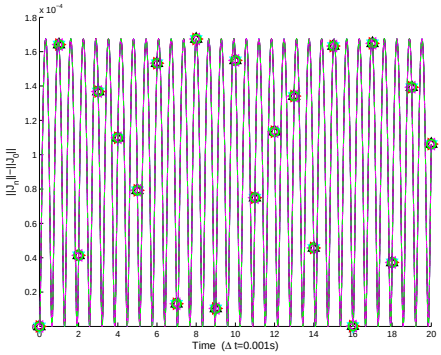
(e) Velocity Norm (f) Acceleration Norm

Figure 8.52: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option I) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]

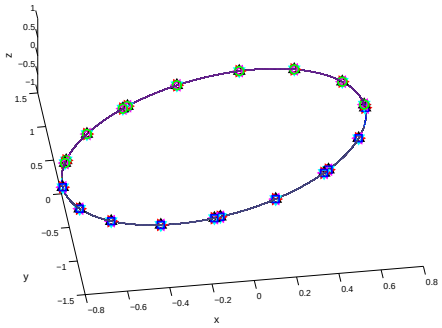


(a) Energy

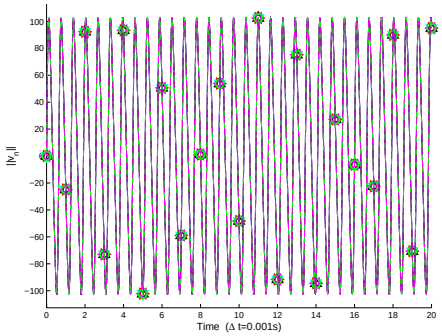
(b) Linear Momentum Ratio



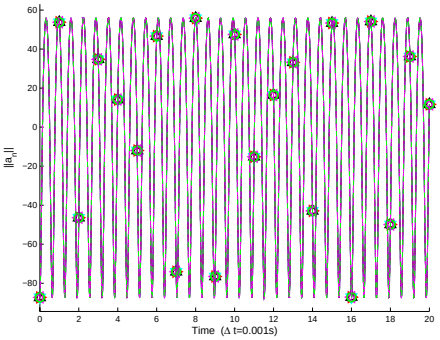
(c) Angular Momentum Difference



(d) Configuration

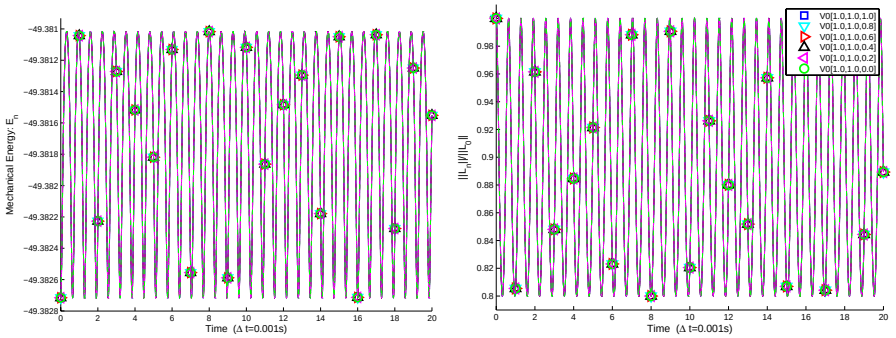


(e) Velocity Norm

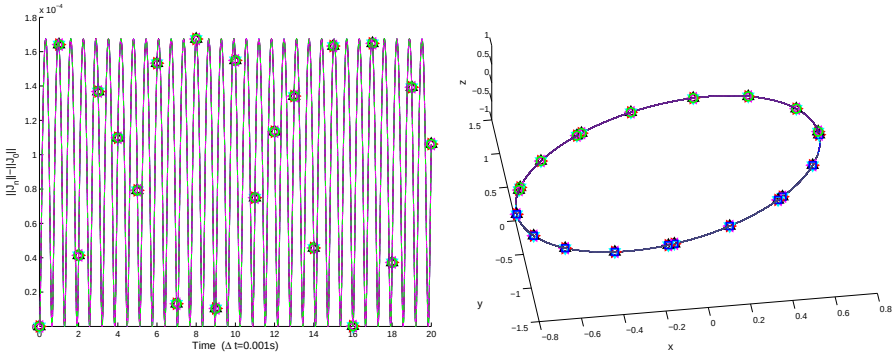


(f) Acceleration Norm

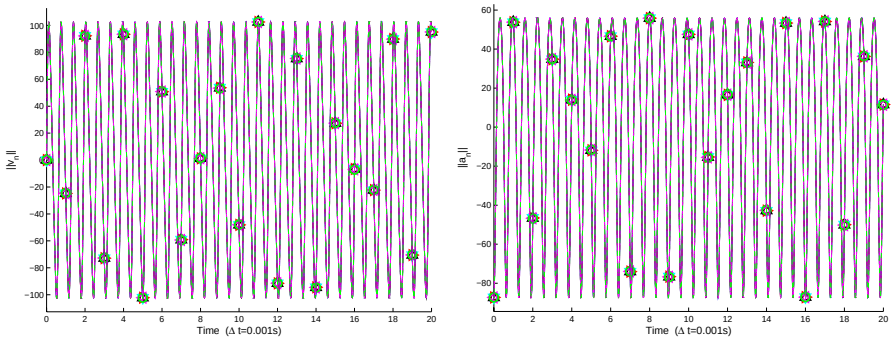
Figure 8.53: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option I) - U0V0(1.0,1.0,ρ_∞)]



(a) Energy (b) Linear Momentum Ratio

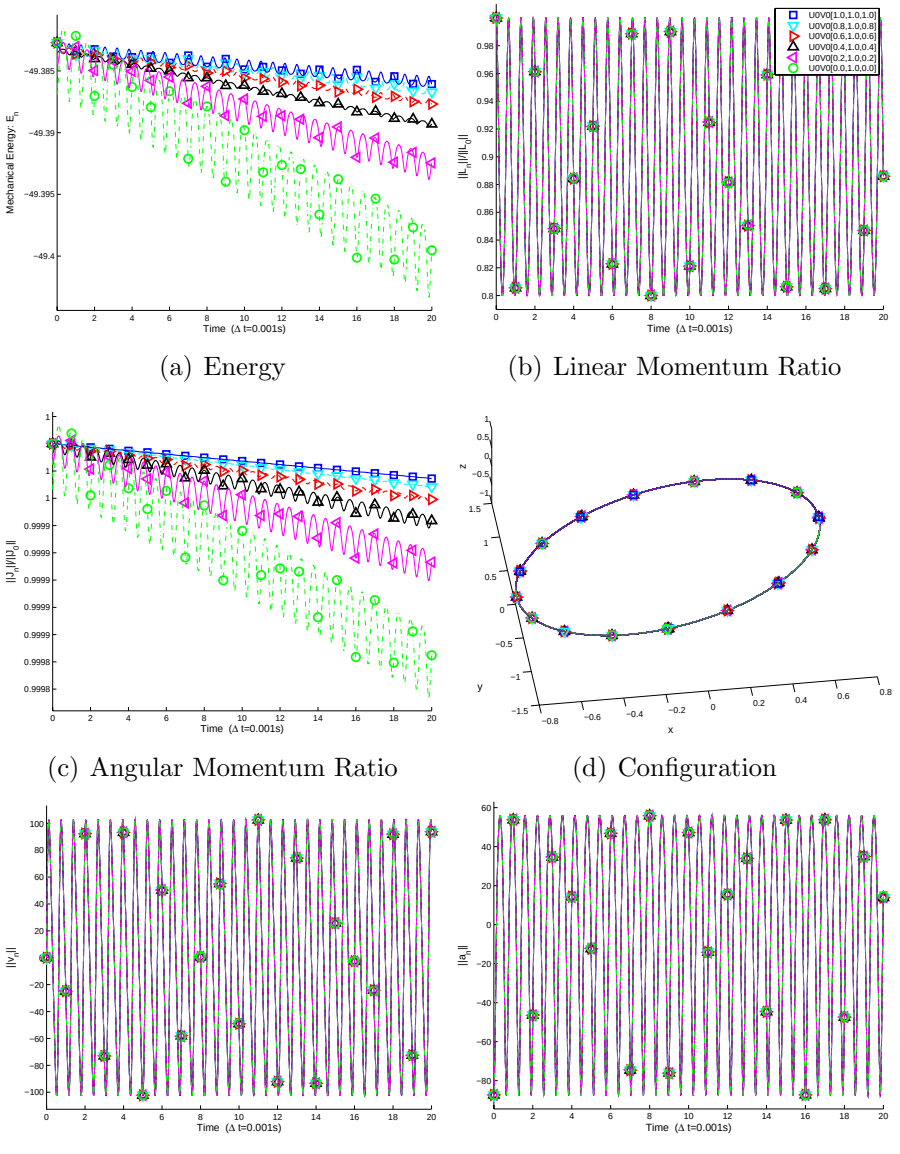


(c) Angular Momentum Difference (d) Configuration



(e) Velocity Norm (f) Acceleration Norm

Figure 8.54: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option I) - V0U0(1.0,1.0,ρ_∞)]

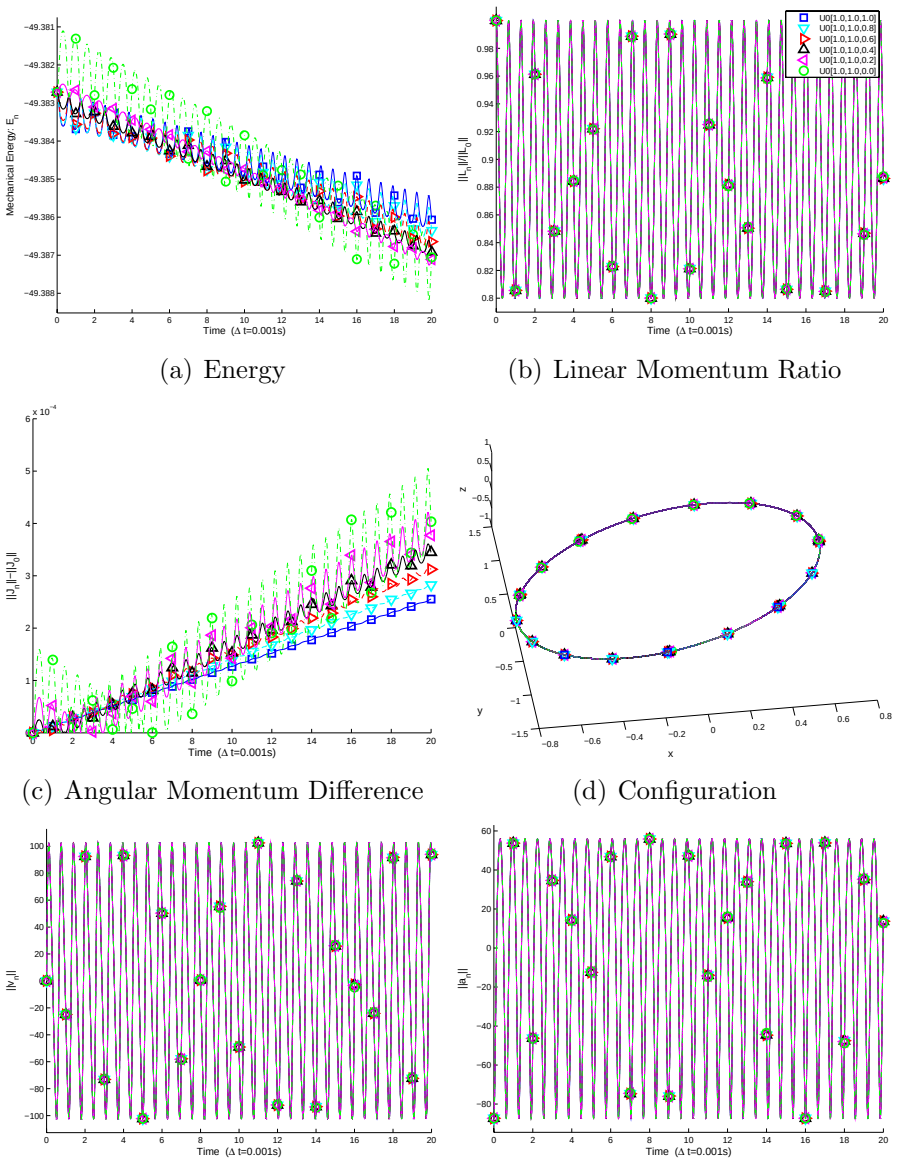


(a) Energy (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.55: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option II) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]

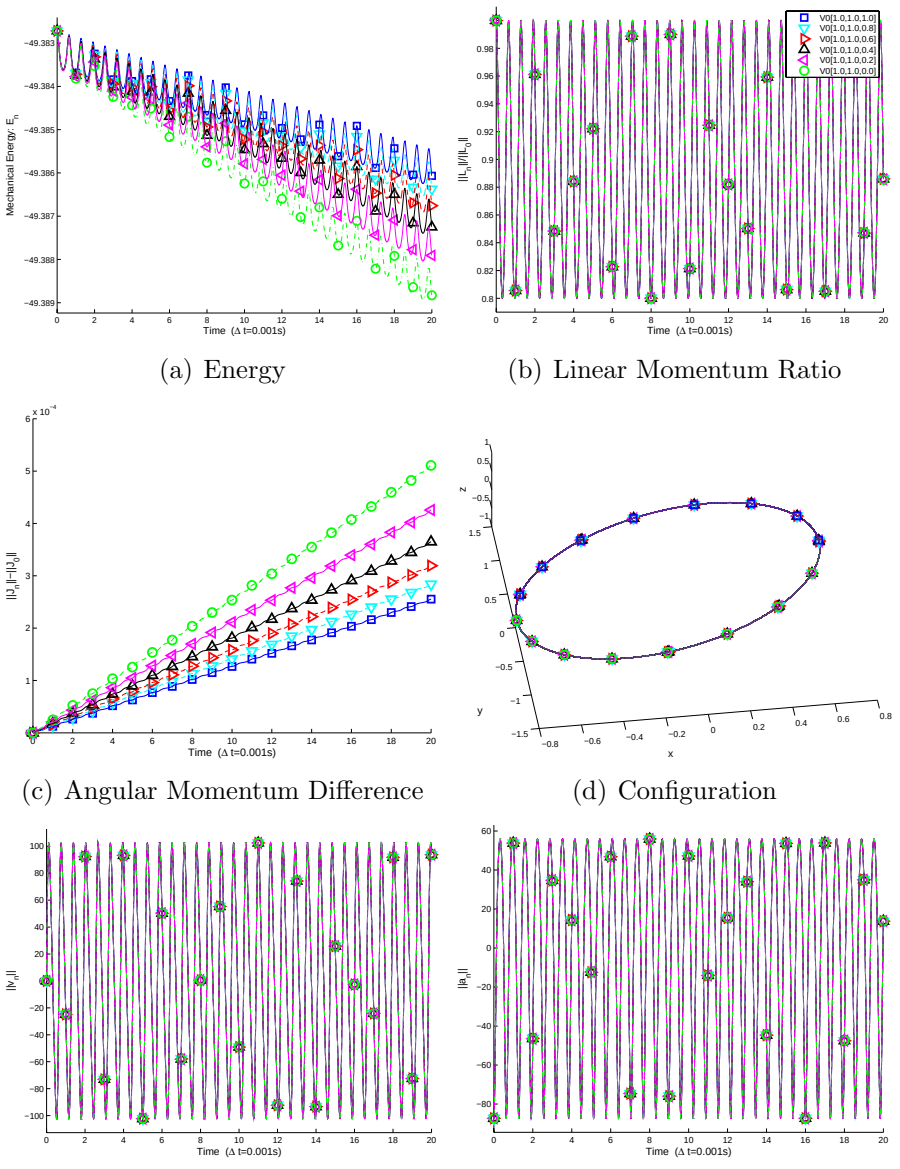


(a) Energy (b) Linear Momentum Ratio

(c) Angular Momentum Difference (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.56: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option II) - U0V0(1.0,1.0,ρ_∞)]

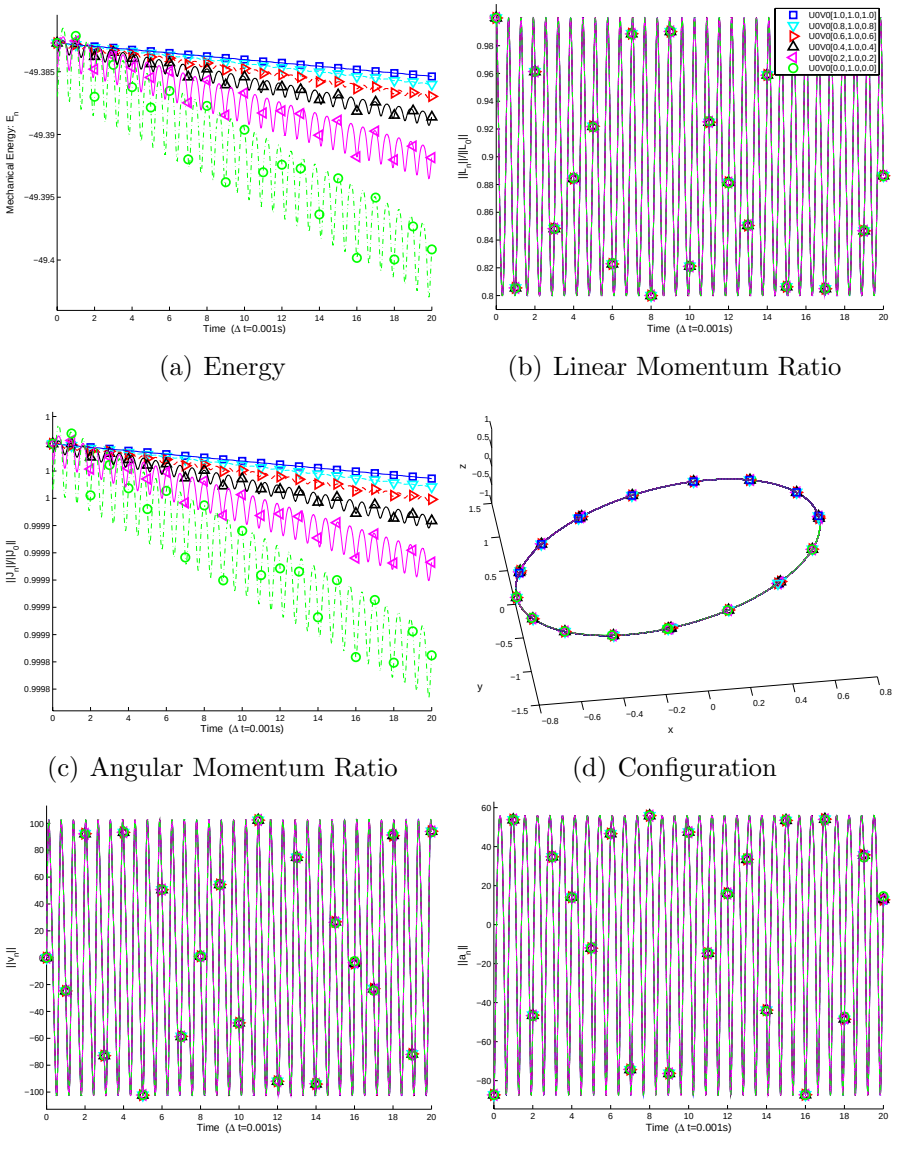


(a) Energy (b) Linear Momentum Ratio

(c) Angular Momentum Difference (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.57: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option II) - V0U0(1.0,1.0,ρ_∞)]

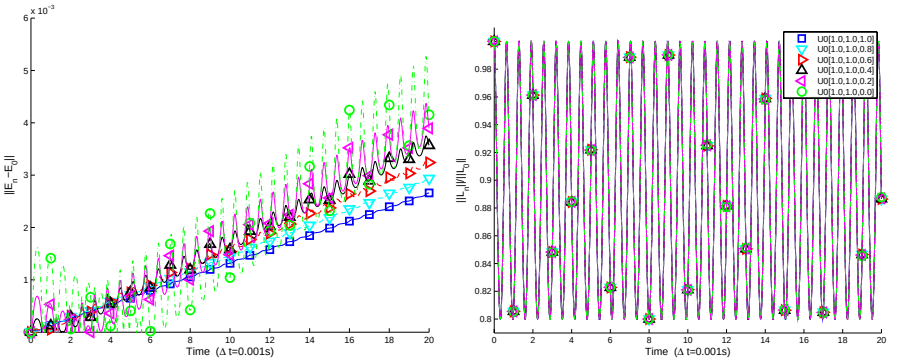


(a) Energy (b) Linear Momentum Ratio

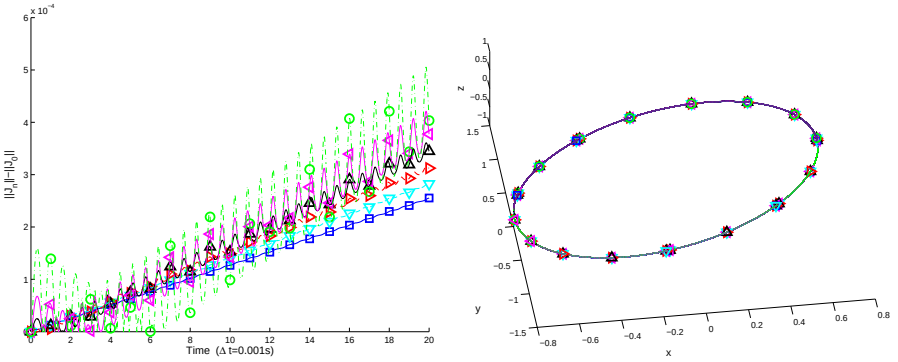
(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

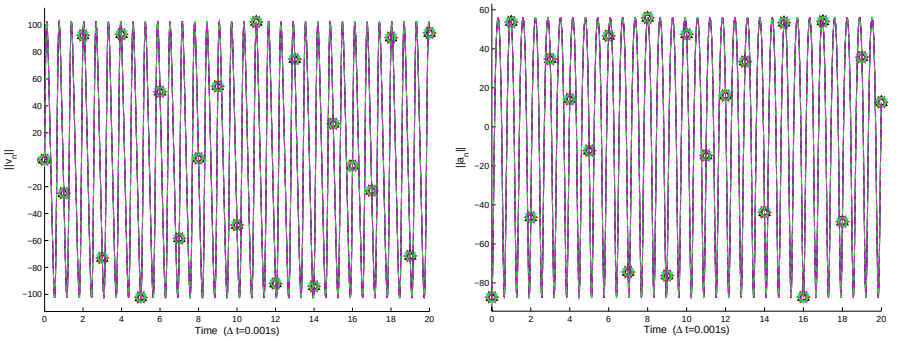
Figure 8.58: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option III) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Energy Difference (b) Linear Momentum Ratio

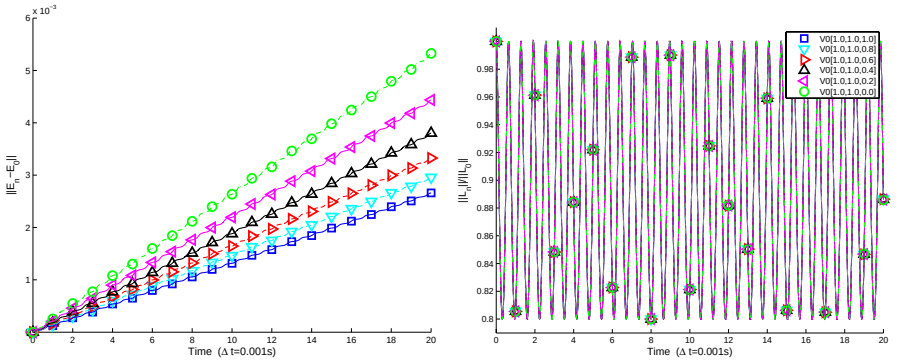


(c) Angular Momentum Difference (d) Configuration

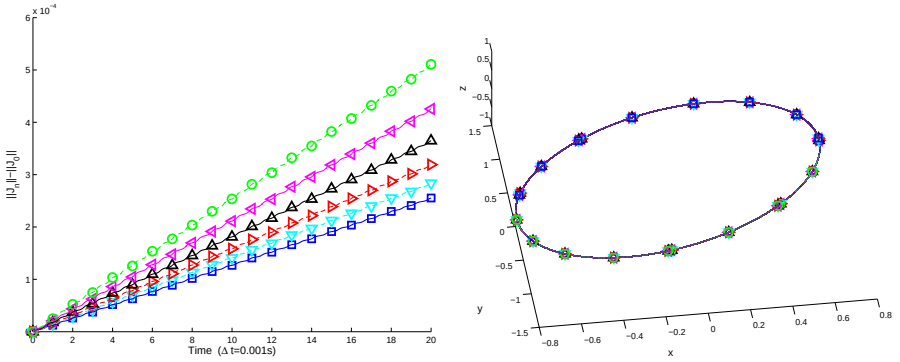


(e) Velocity Norm (f) Acceleration Norm

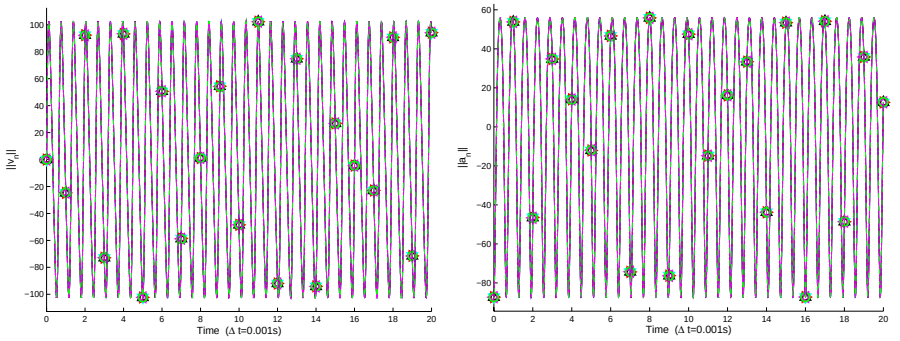
Figure 8.59: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option III) - U0V0(1.0,1.0,ρ_∞)]



(a) Energy Difference (b) Linear Momentum Ratio

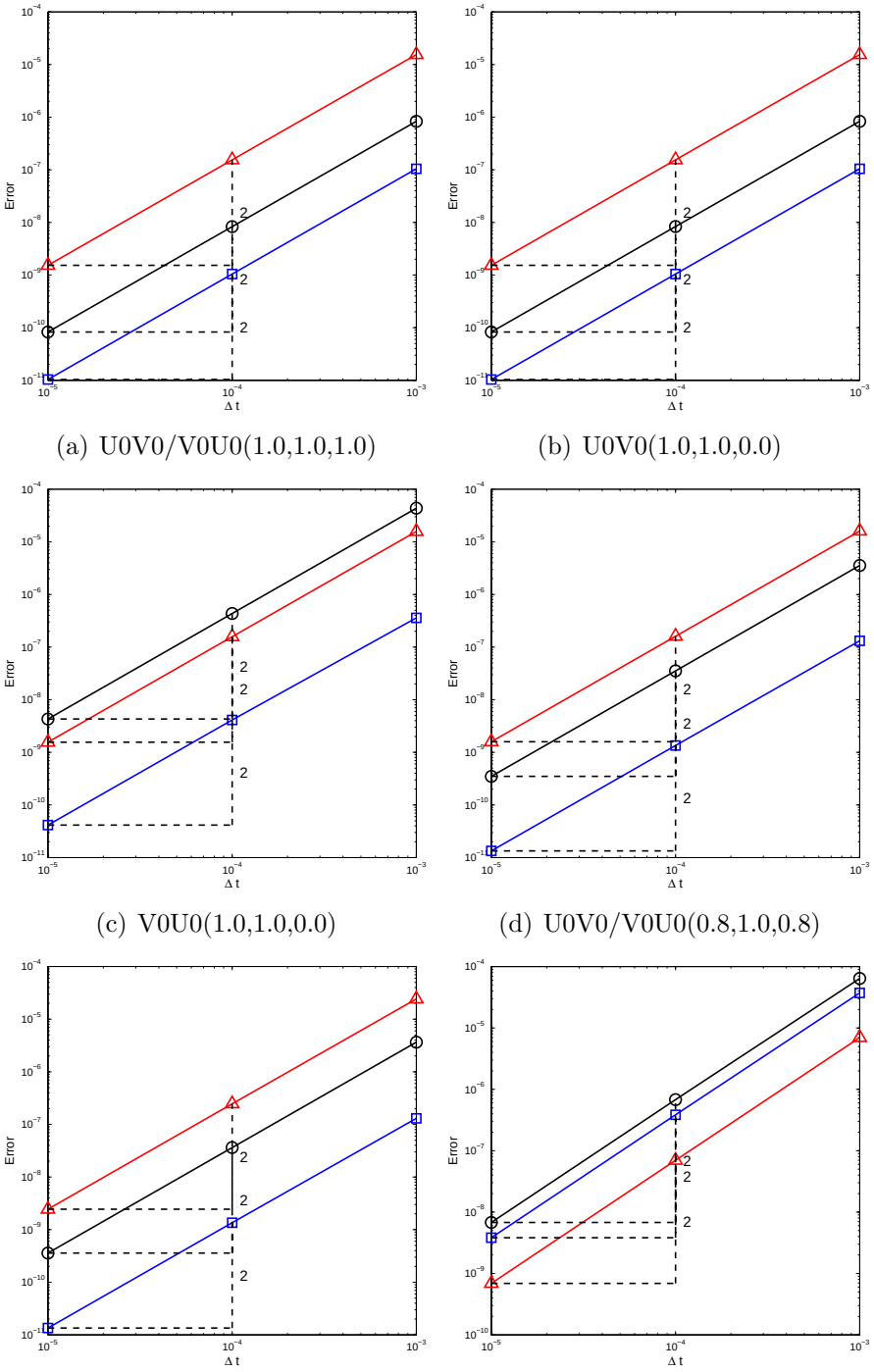


(c) Angular Momentum Difference (d) Configuration



(e) Velocity Norm (f) Acceleration Norm

Figure 8.60: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option III) - V0U0(1.0,1.0,ρ_∞)]



(a) U0V0/V0U0(1.0,1.0,1.0)

(b) U0V0(1.0,1.0,0.0)

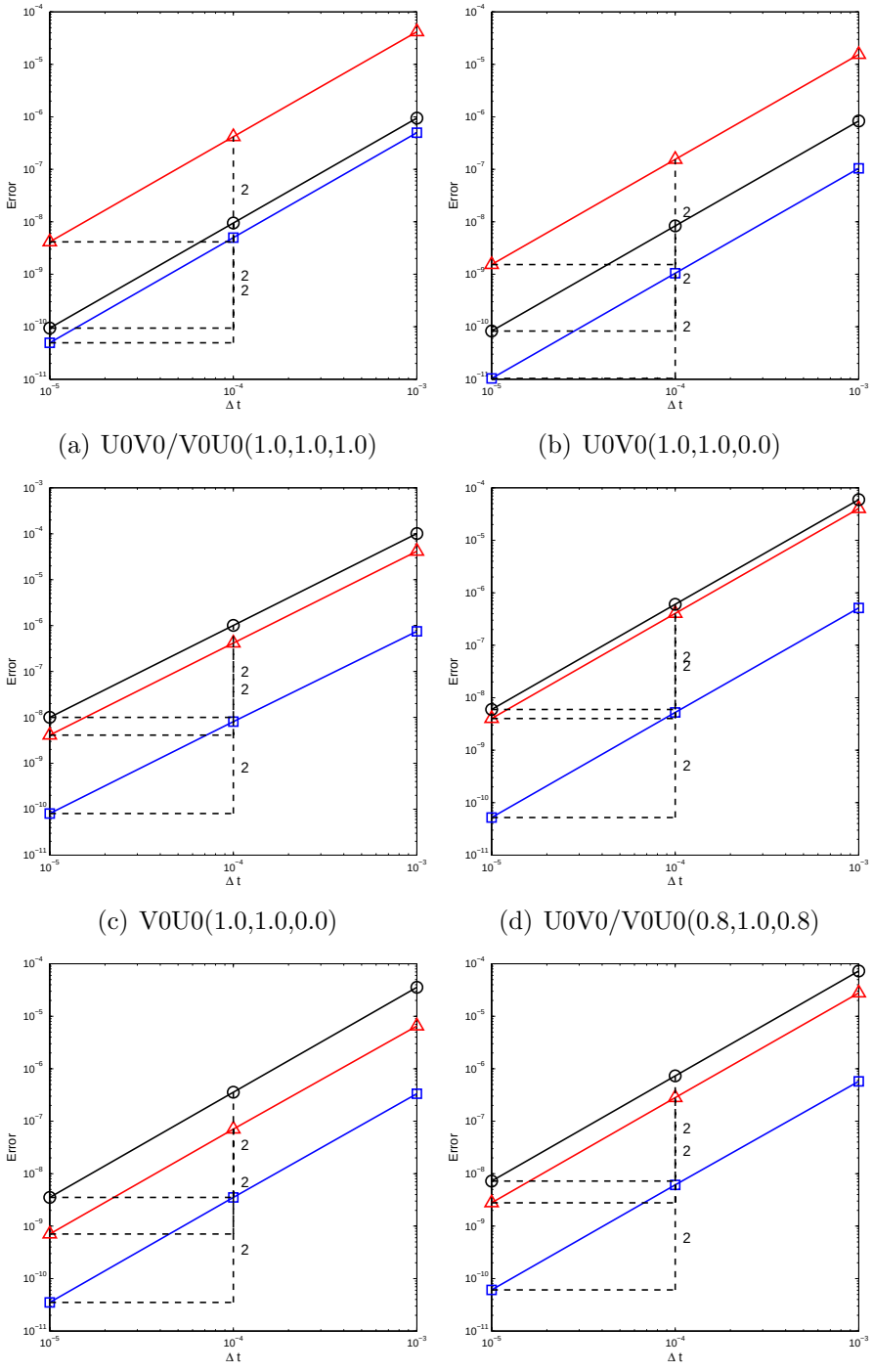
(c) V0U0(1.0,1.0,0.0)

(d) U0V0/V0U0(0.8,1.0,0.8)

(e) U0V1(0.6,0.9,0.2)

(f) V0U1(0.6,0.9,0.2)

Figure 8.61: Time accuracies in the configuration (□), velocity (△), and acceleration (○) in the forced mechanical system. [Problem: Kepler's problem] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 0$ (Option I)]



(a) U0V0/V0U0(1.0,1.0,1.0)

(b) U0V0(1.0,1.0,0.0)

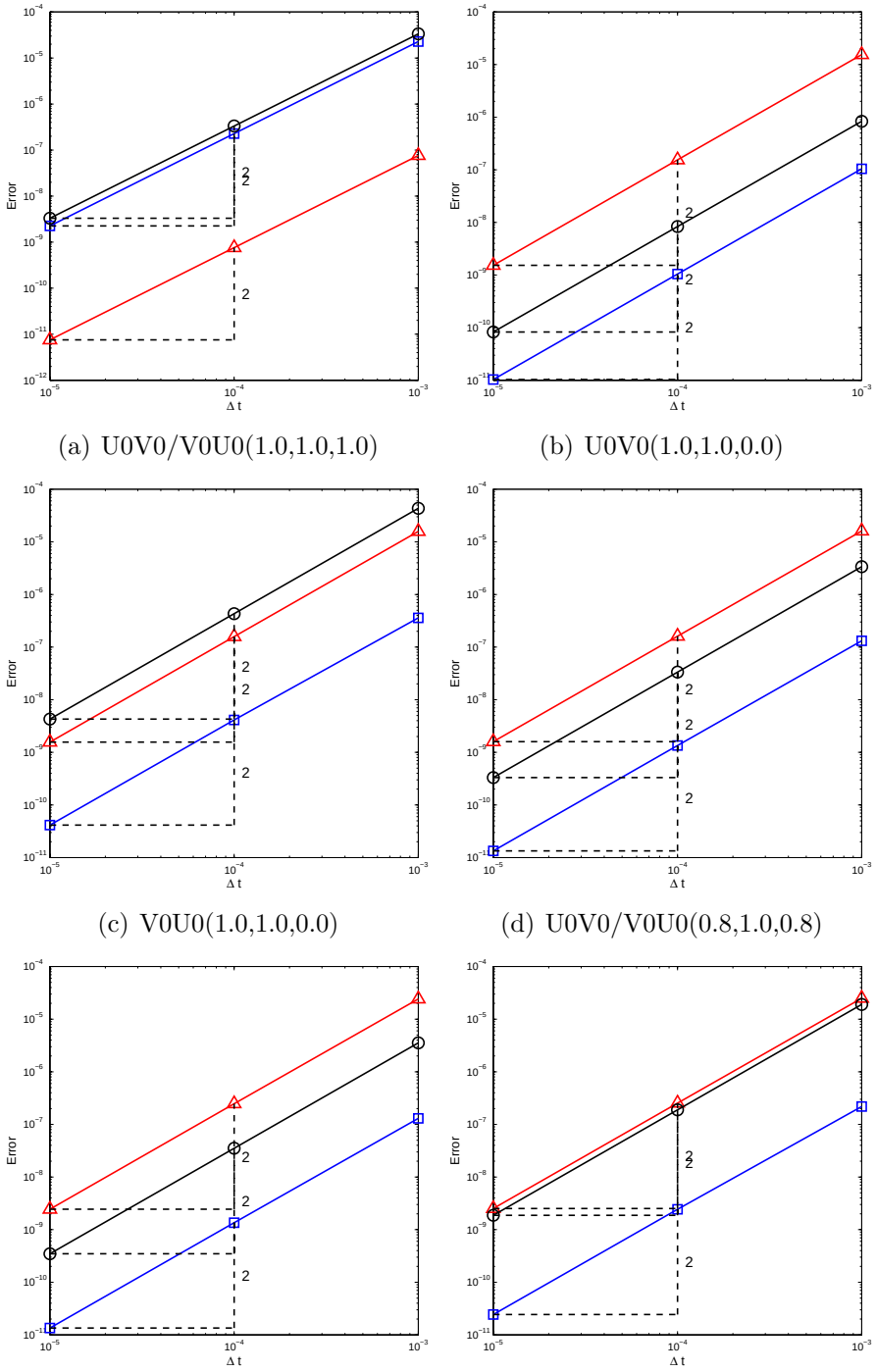
(c) V0U0(1.0,1.0,0.0)

(d) U0V0/V0U0(0.8,1.0,0.8)

(e) U0V1(0.6,0.9,0.2)

(f) V0U1(0.6,0.9,0.2)

Figure 8.62: Time accuracies in the configuration (□), velocity (△), and acceleration (○) in the forced mechanical system [Problem: Kepler's problem] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 0$ (Option II)]



(a) U0V0/V0U0(1.0,1.0,1.0)

(b) U0V0(1.0,1.0,0.0)

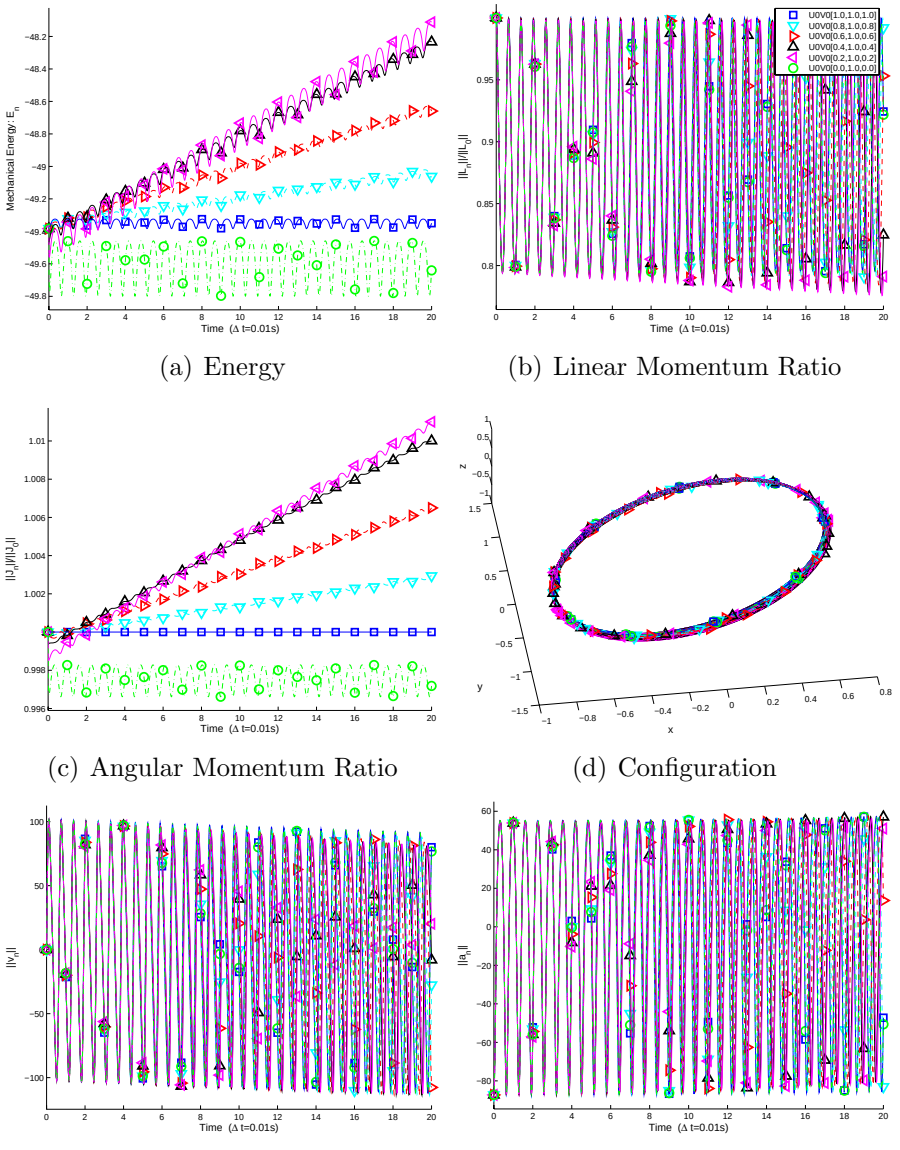
(c) V0U0(1.0,1.0,0.0)

(d) U0V0/V0U0(0.8,1.0,0.8)

(e) U0V1(0.6,0.9,0.2)

(f) V0U1(0.6,0.9,0.2)

Figure 8.63: Time accuracies in the configuration, velocity, and acceleration in the forced mechanical system. [Problem: Kepler's problem] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 0$ (Option III)]

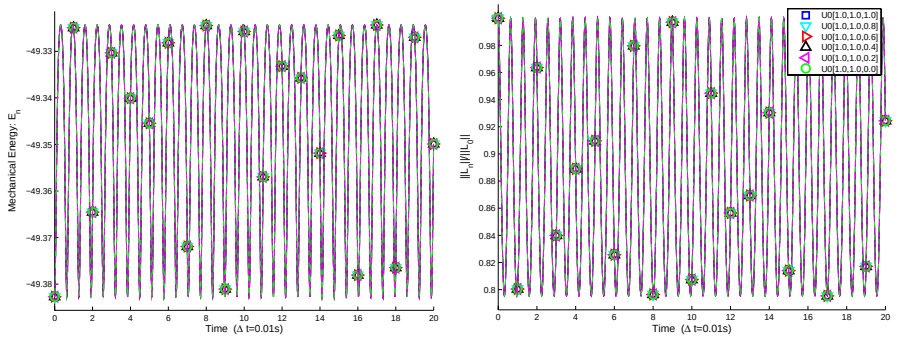


(a) Energy (b) Linear Momentum Ratio

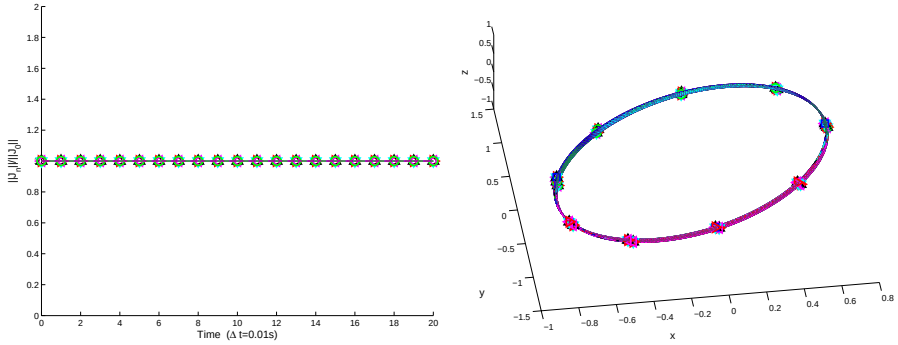
(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

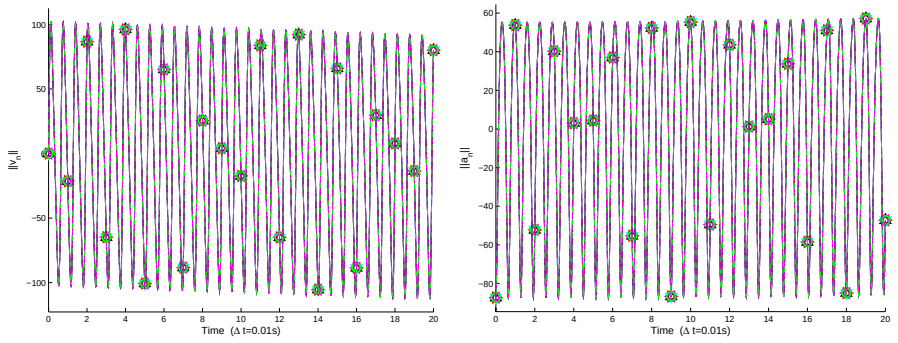
Figure 8.64: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option I) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Energy (b) Linear Momentum Ratio

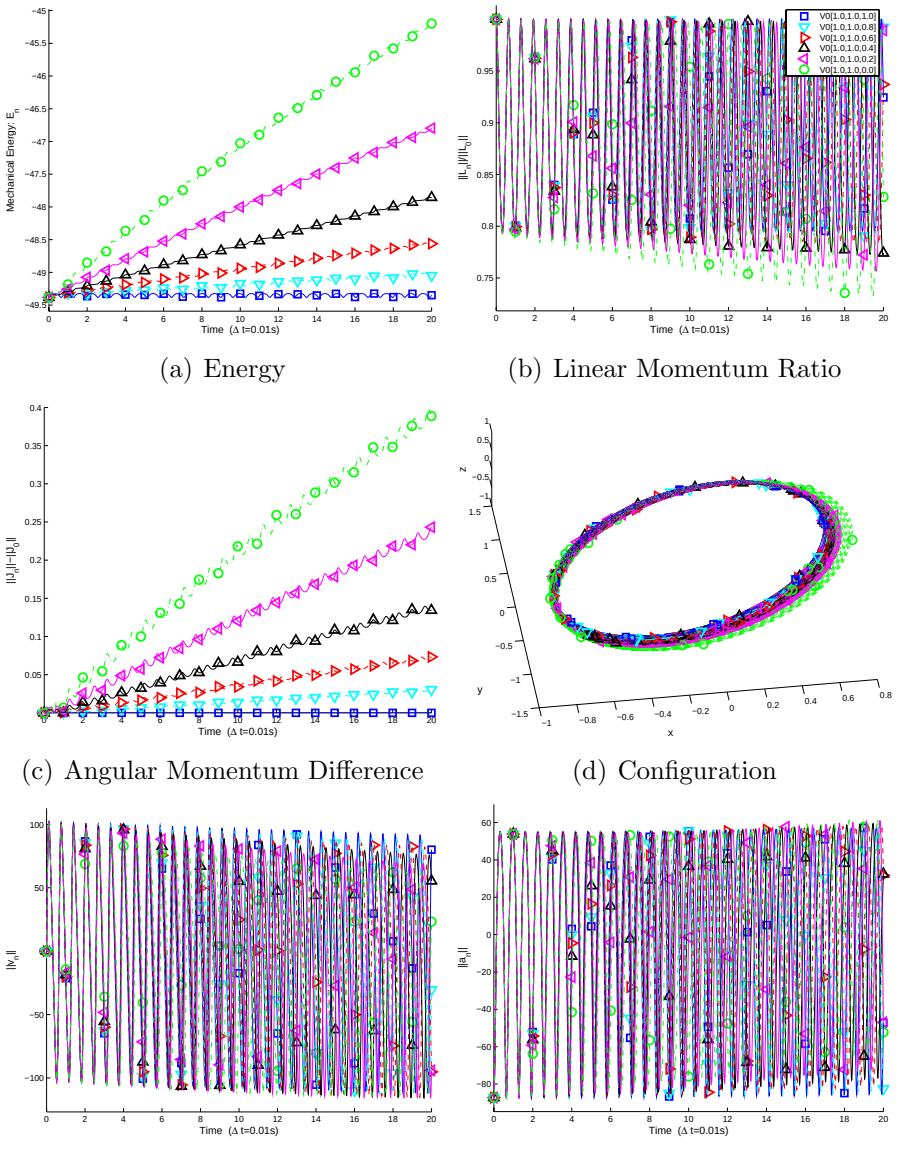


(c) Angular Momentum Ratio (d) Configuration



(e) Velocity Norm (f) Acceleration Norm

Figure 8.65: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option I) - U0V0(1.0,1.0,ρ_∞)]

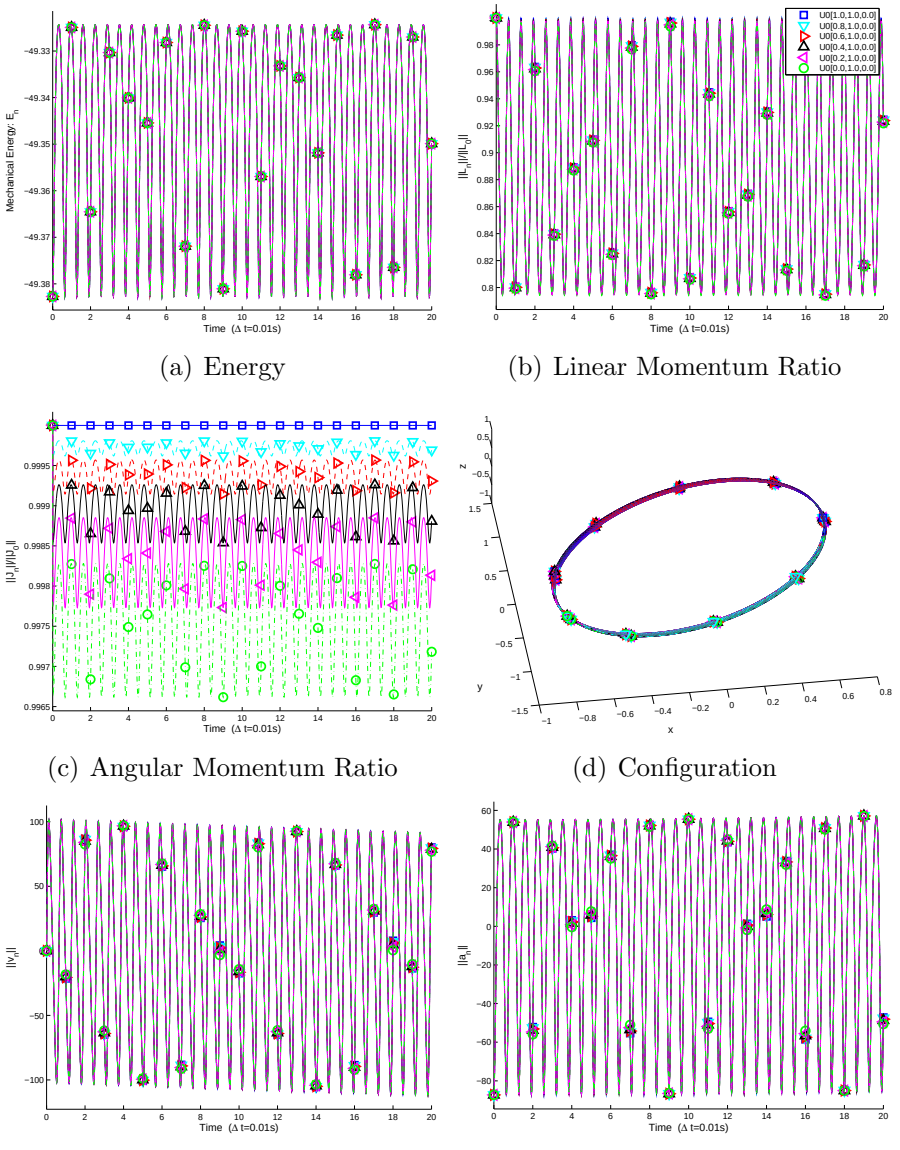


(a) Energy (b) Linear Momentum Ratio

(c) Angular Momentum Difference (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.66: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option I) - V0U0(1.0,1.0,ρ_∞)]

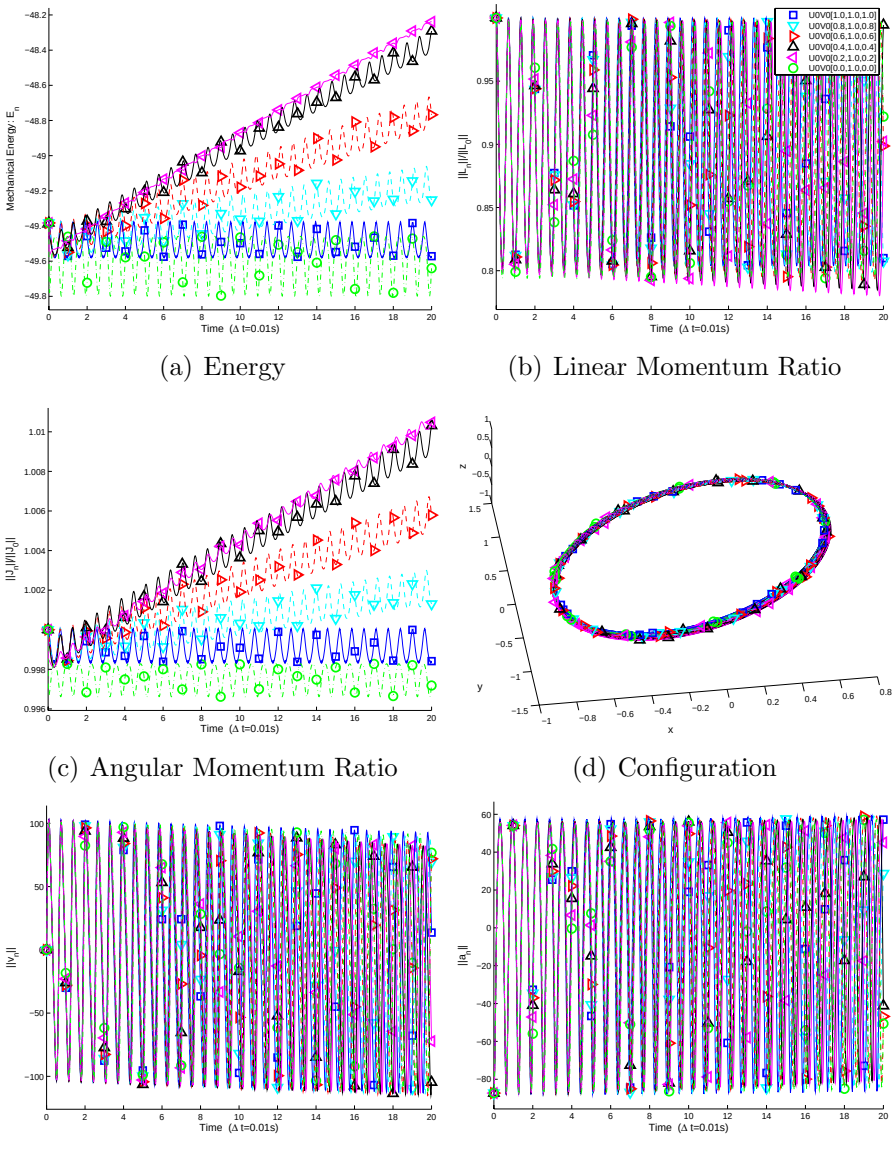


(a) Energy (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.67: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option I) - U0(ρ_∞,1.0,0.0)]

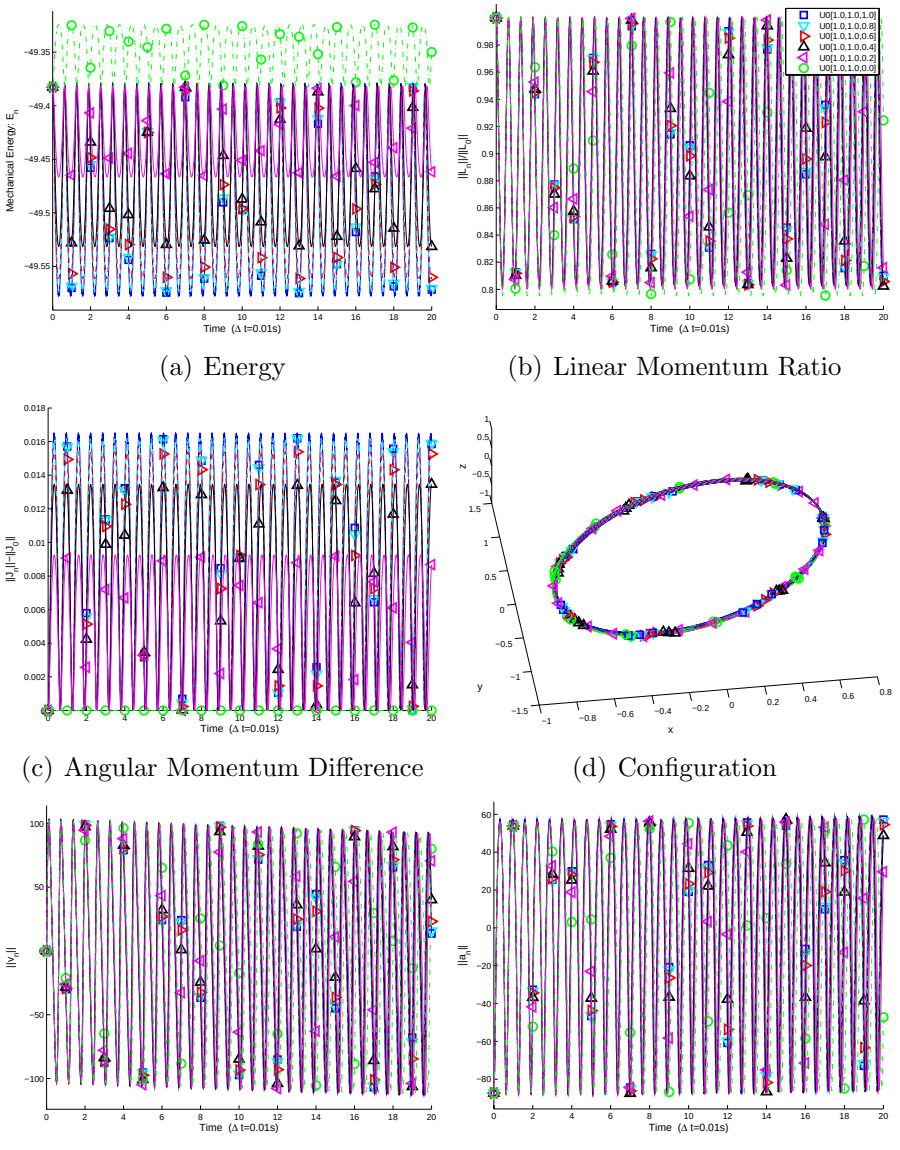


(a) Energy (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.68: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option II) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]

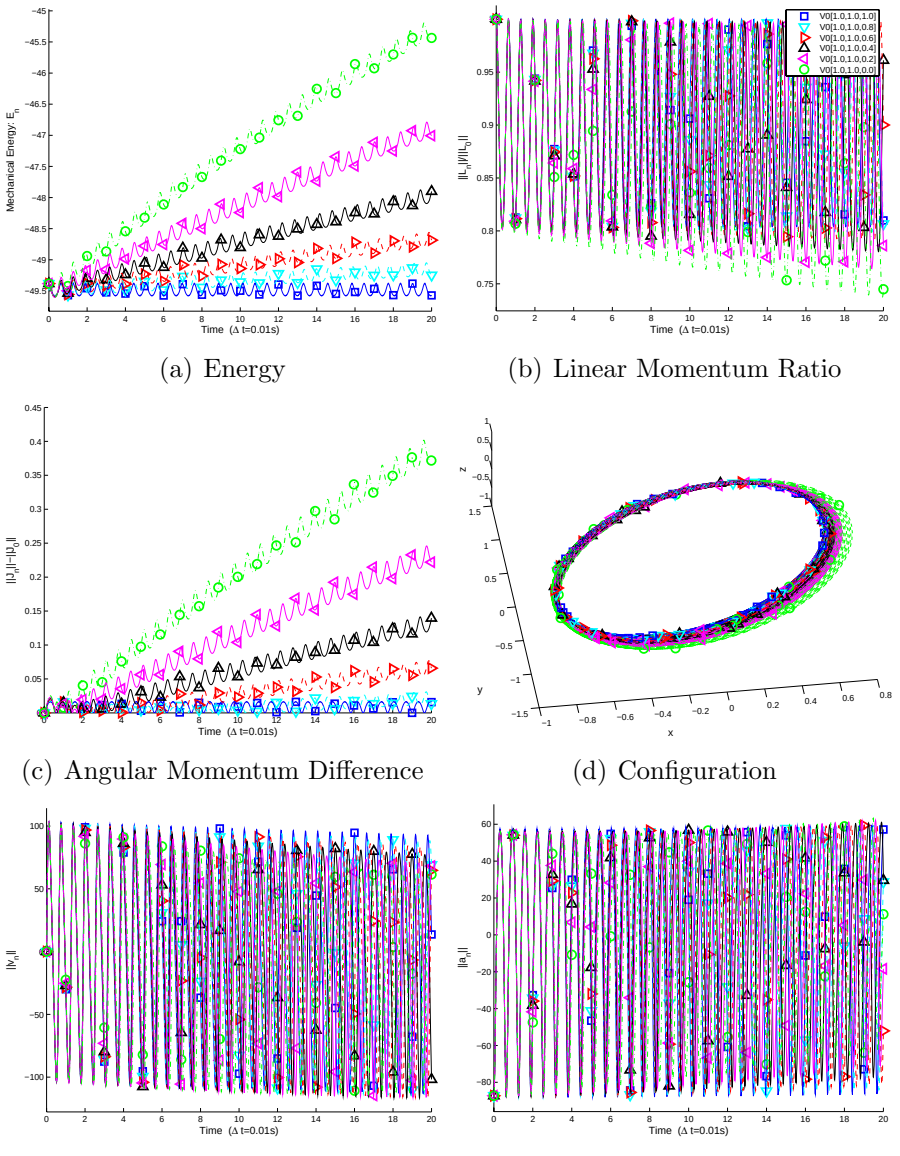


(a) Energy (b) Linear Momentum Ratio

(c) Angular Momentum Difference (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.69: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option II) - U0V0(1.0,1.0,ρ_∞)]

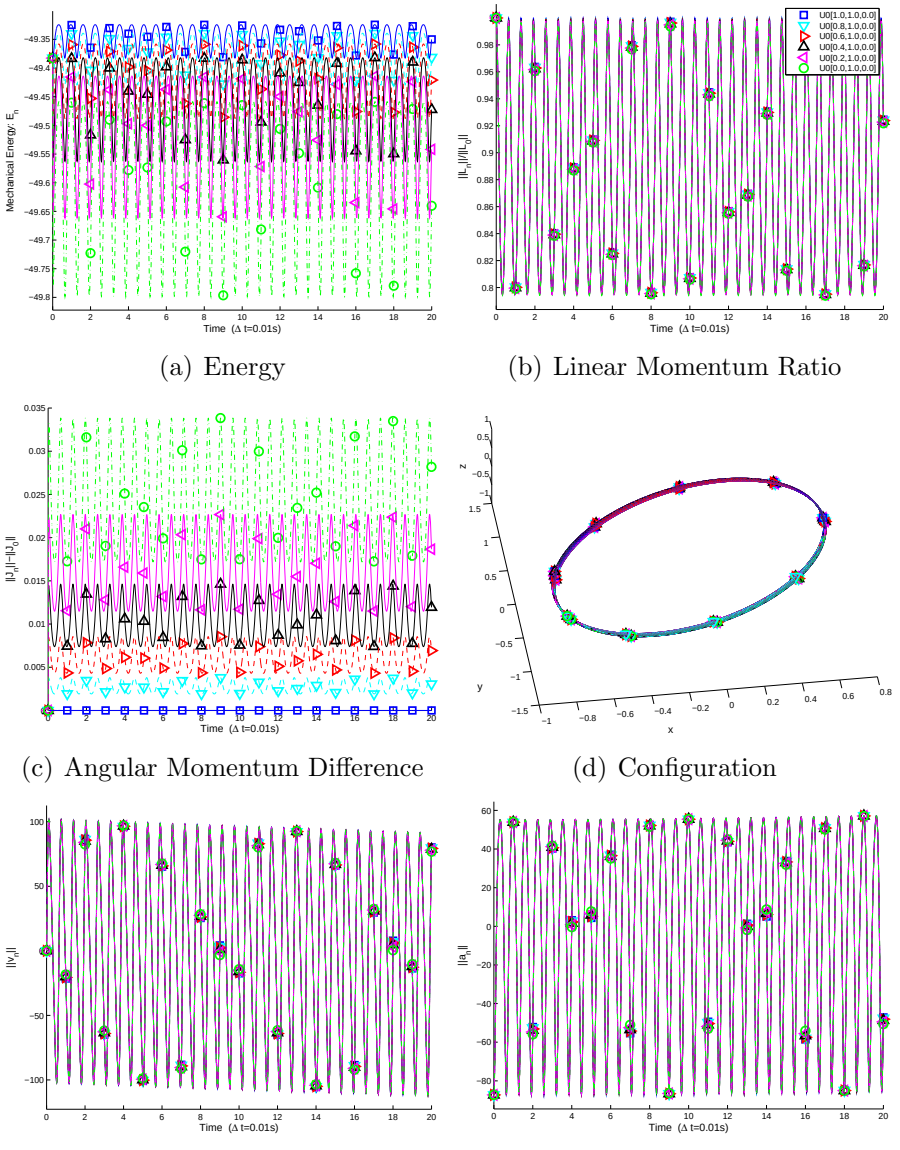


(a) Energy (b) Linear Momentum Ratio

(c) Angular Momentum Difference (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.70: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option II) - V0U0(1.0,1.0,ρ_∞)]

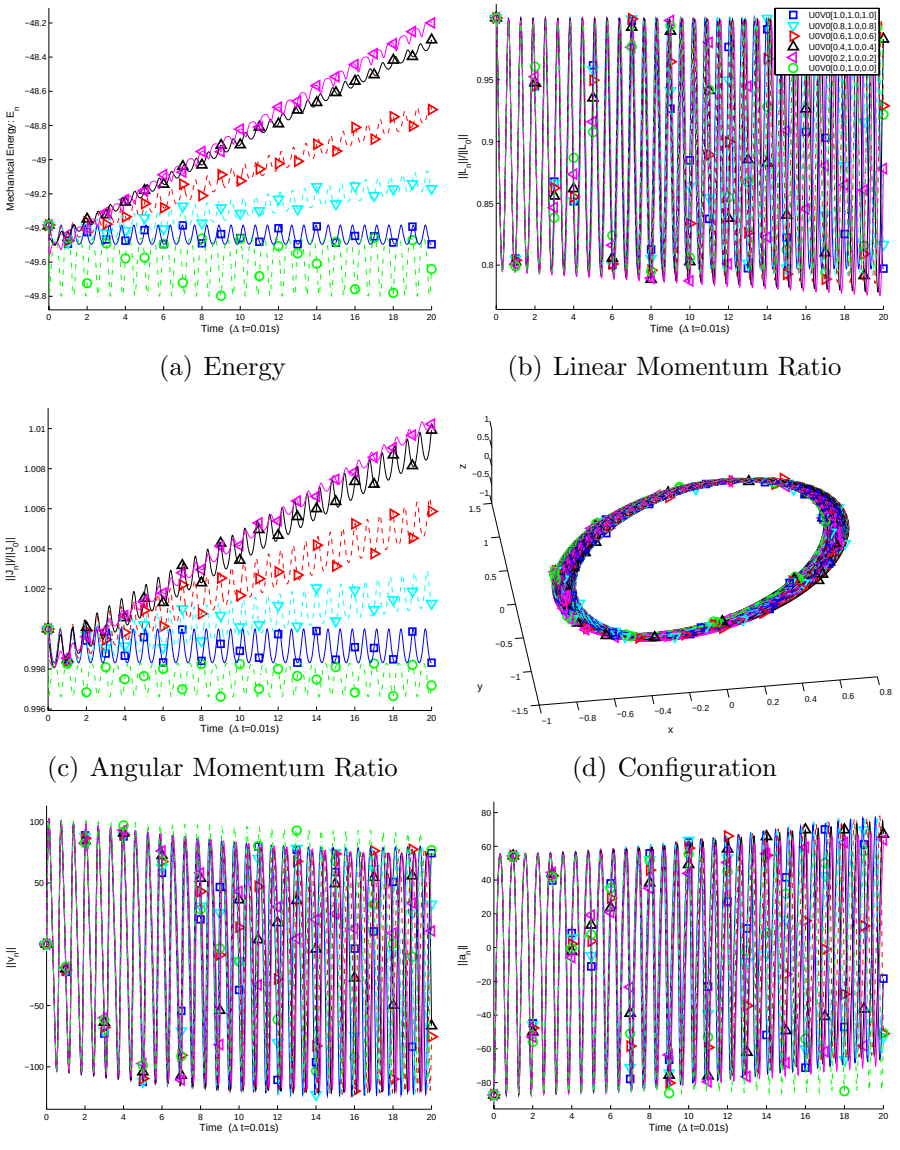


(a) Energy (b) Linear Momentum Ratio

(c) Angular Momentum Difference (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.71: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option II) - U0(ρ_∞,1.0,0.0)]

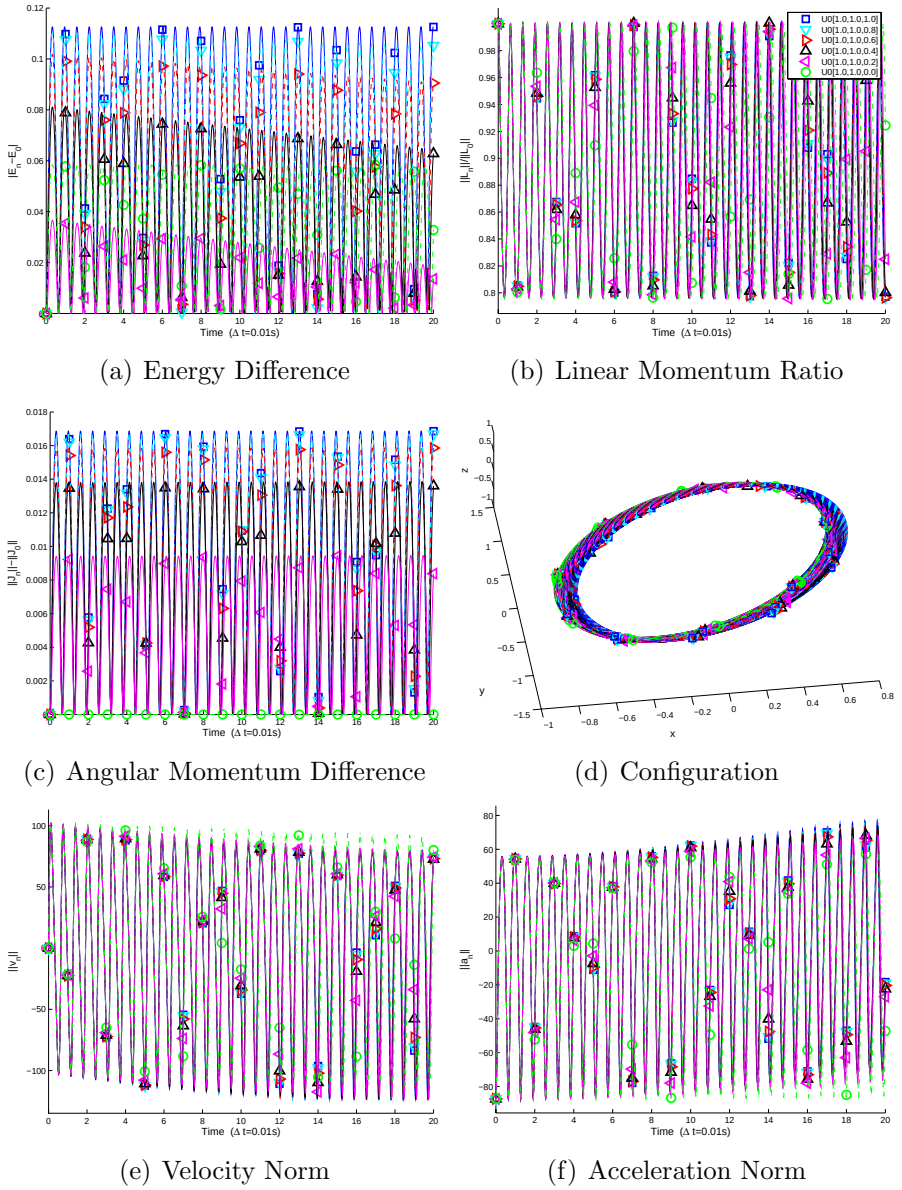


(a) Energy (b) Linear Momentum Ratio

(c) Angular Momentum Ratio (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.72: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option III) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]

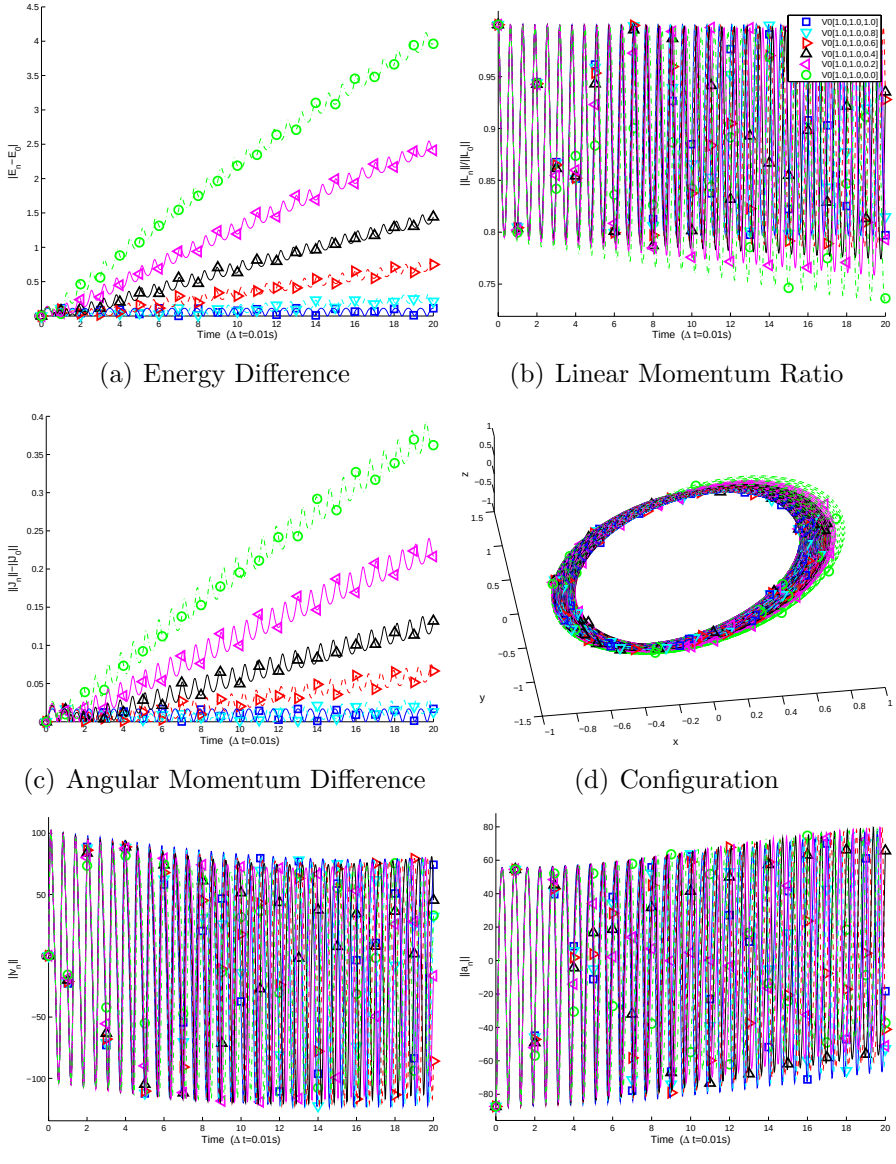


(a) Energy Difference (b) Linear Momentum Ratio

(c) Angular Momentum Difference (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

Figure 8.73: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option III) - U0V0(1.0,1.0,ρ_∞)]

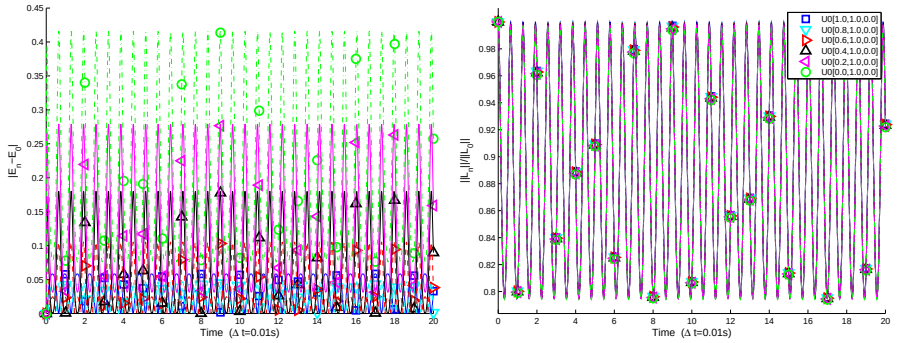


(a) Energy Difference (b) Linear Momentum Ratio

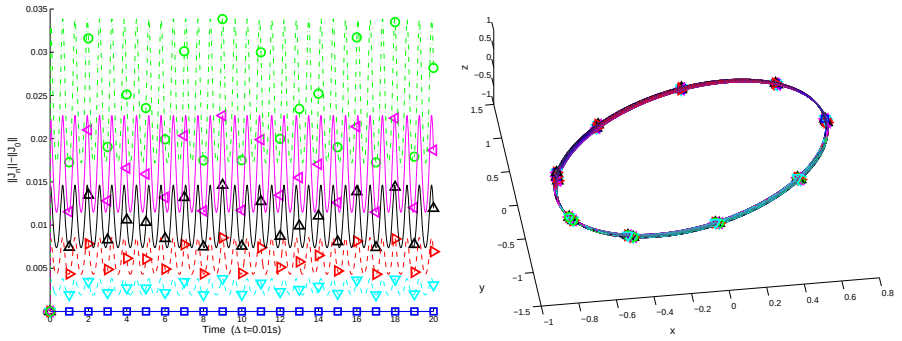
(c) Angular Momentum Difference (d) Configuration

(e) Velocity Norm (f) Acceleration Norm

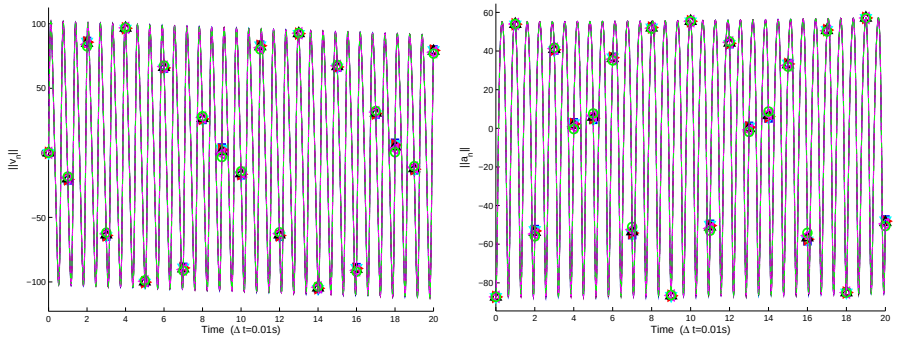
Figure 8.74: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option III) - V0U0(1.0,1.0,ρ_∞)]



(a) Energy Difference (b) Linear Momentum Ratio

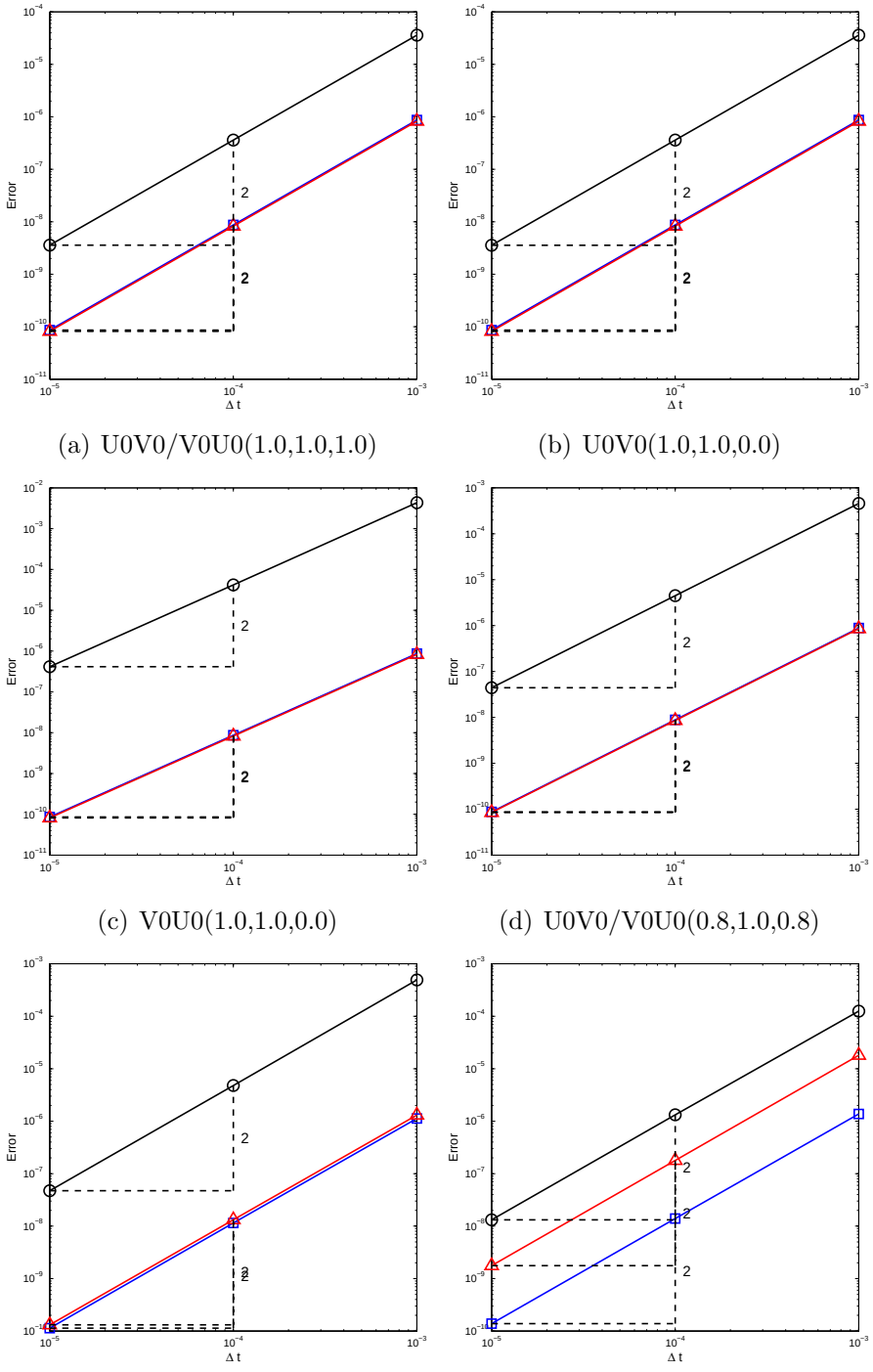


(c) Angular Momentum Difference (d) Configuration



(e) Velocity Norm (f) Acceleration Norm

Figure 8.75: Time histories in the ***conservative system***. [Problem: Kepler's problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option III) - U0(ρ_∞,1.0,0.0)]

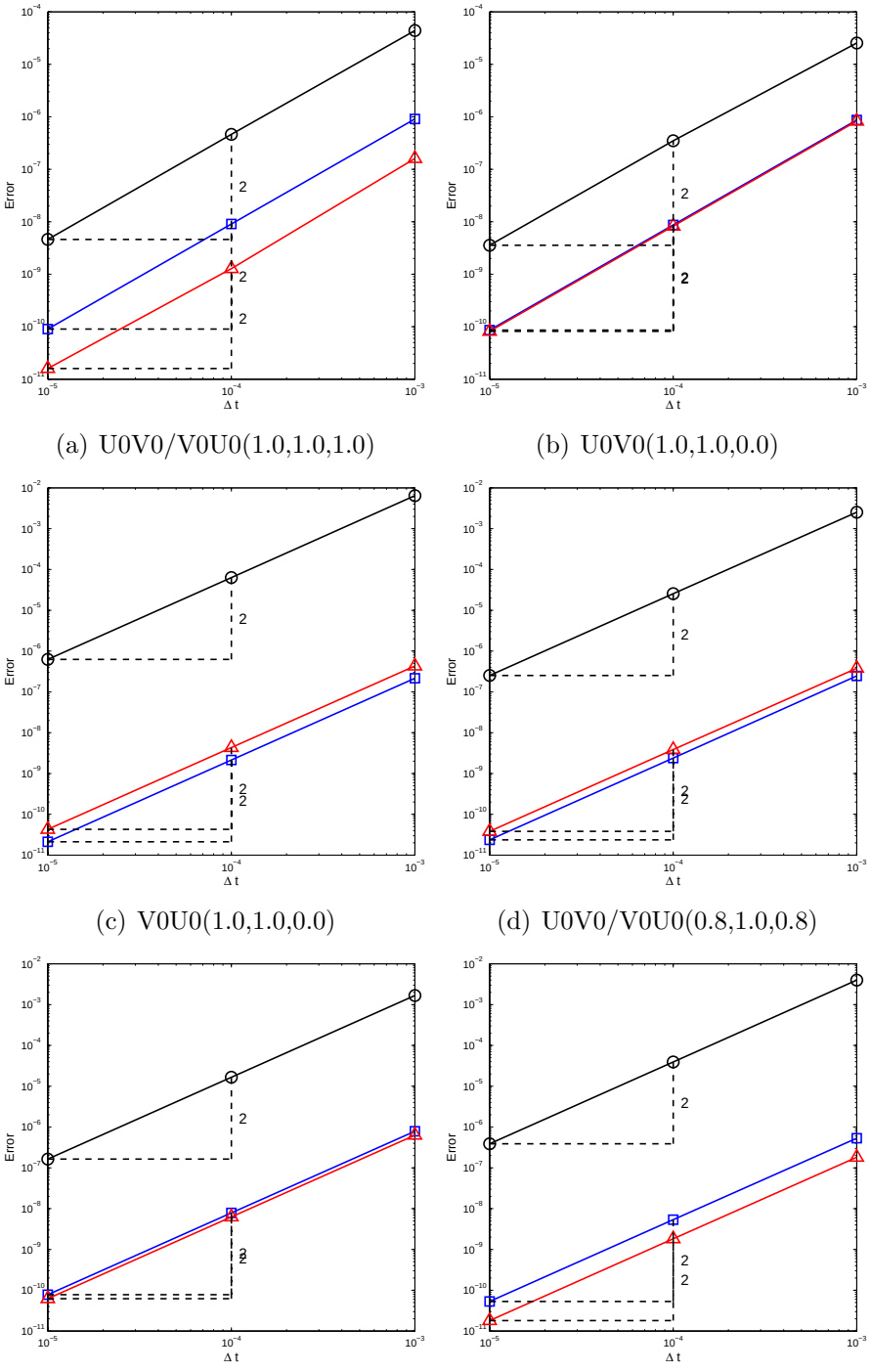


(a) U0V0/V0U0(1.0,1.0,1.0) (b) U0V0(1.0,1.0,0.0)

(c) V0U0(1.0,1.0,0.0) (d) U0V0/V0U0(0.8,1.0,0.8)

(e) U0V1(0.6,0.9,0.2) (f) V0U1(0.6,0.9,0.2)

Figure 8.76: Time accuracies in the configuration (□), velocity (△), and acceleration (○) in the forced mechanical system. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 1$ (Option I)]



(a) U0V0/V0U0(1.0,1.0,1.0)

(b) U0V0(1.0,1.0,0.0)

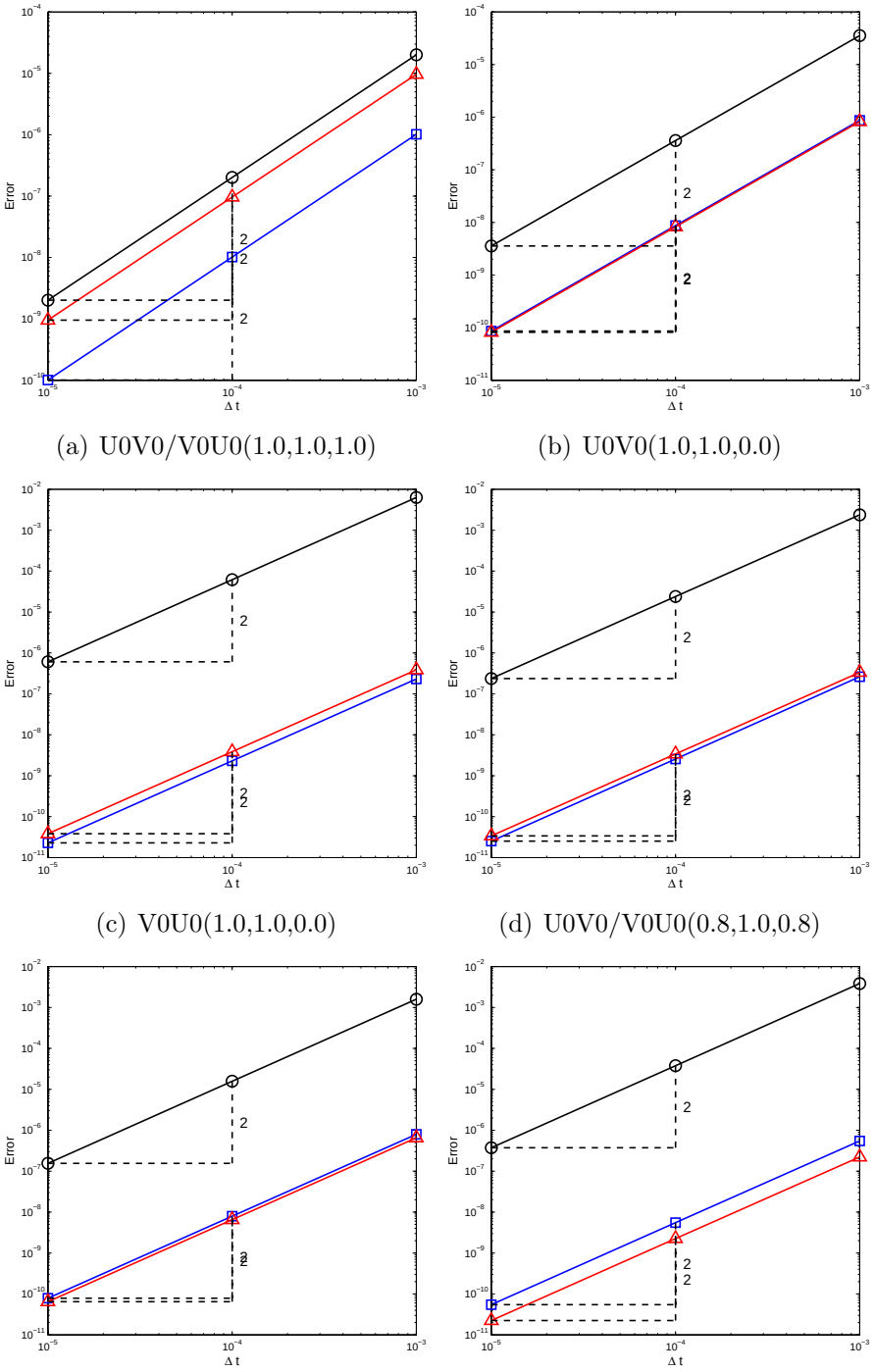
(c) V0U0(1.0,1.0,0.0)

(d) U0V0/V0U0(0.8,1.0,0.8)

(e) U0V1(0.6,0.9,0.2)

(f) V0U1(0.6,0.9,0.2)

Figure 8.77: Time accuracies in the configuration (□), velocity (△), and acceleration (○) in the forced mechani [Problem: Lennard-Jones potential problem] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 1$ (Option II)]



(a) U0V0/V0U0(1.0,1.0,1.0)

(b) U0V0(1.0,1.0,0.0)

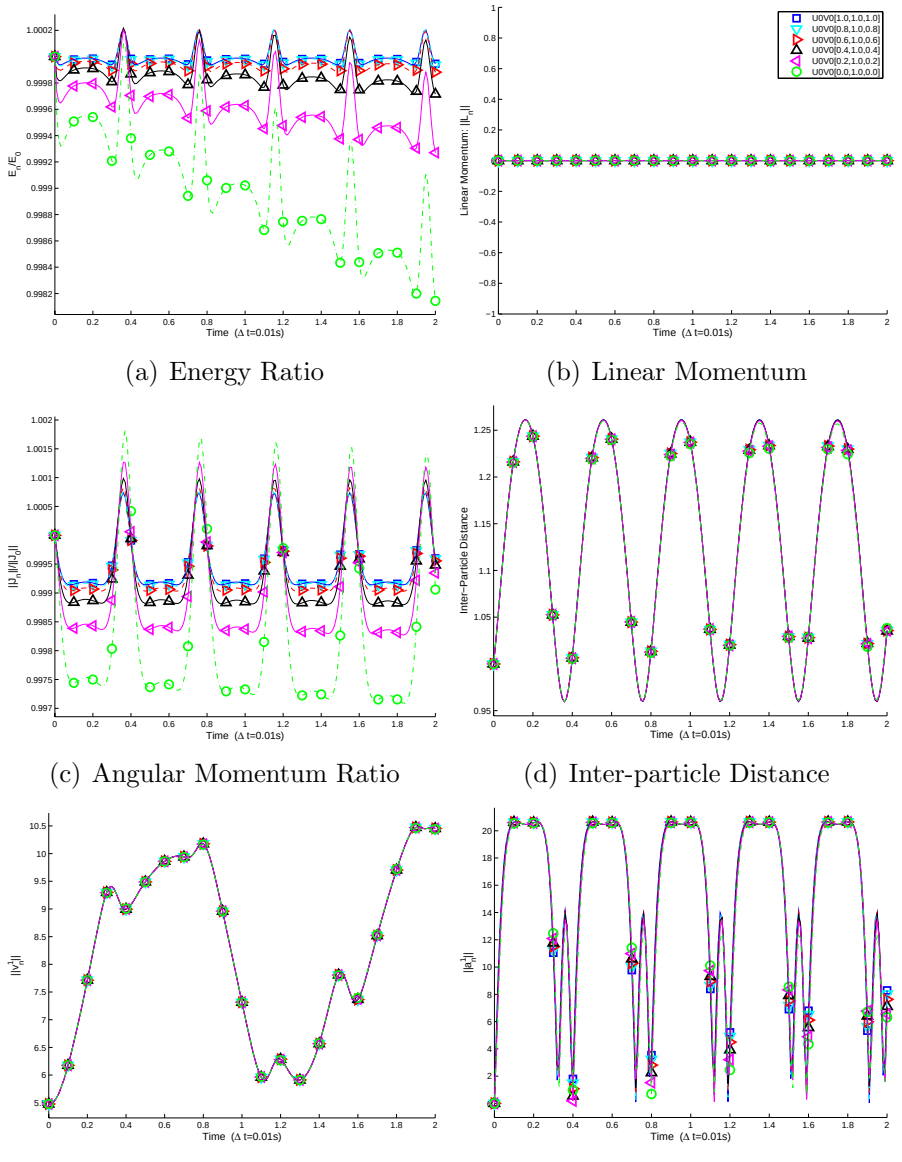
(c) V0U0(1.0,1.0,0.0)

(d) U0V0/V0U0(0.8,1.0,0.8)

(e) U0V1(0.6,0.9,0.2)

(f) V0U1(0.6,0.9,0.2)

Figure 8.78: Time accuracies in the configuration, velocity, and acceleration in the forced mechanical system. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 1$ (Option III)]



(a) Energy Ratio

(b) Linear Momentum

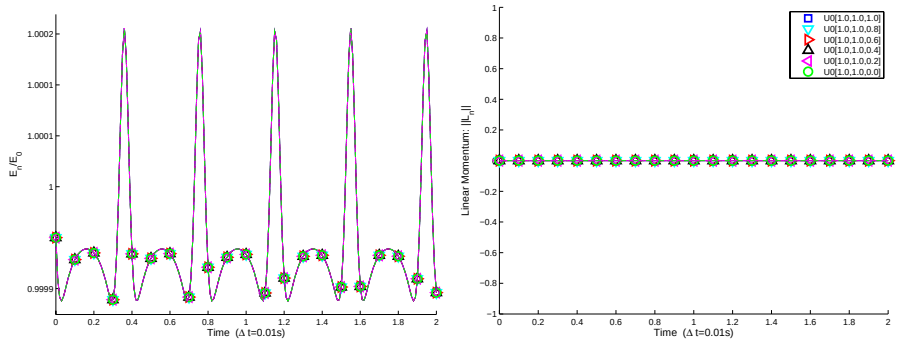
(c) Angular Momentum Ratio

(d) Inter-particle Distance

(e) Velocity Norm

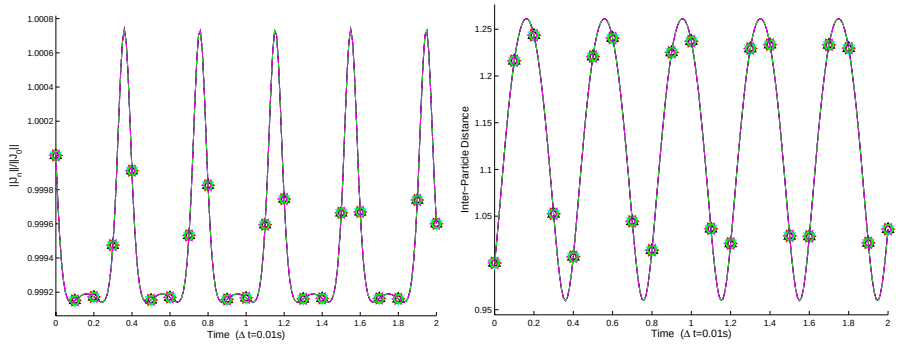
(f) Acceleration Norm

Figure 8.79: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option I) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



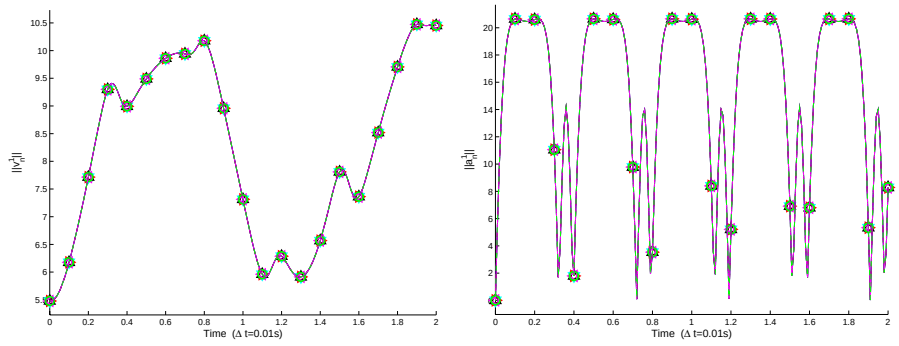
(a) Energy Ratio

(b) Linear Momentum



(c) Angular Momentum Ratio

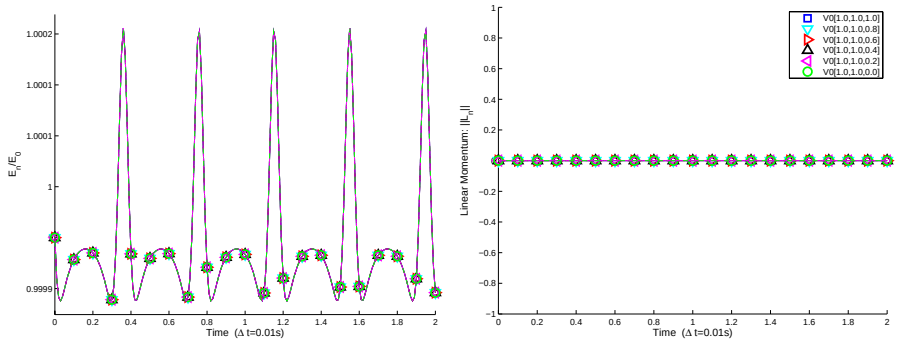
(d) Inter-particle Distance



(e) Velocity Norm

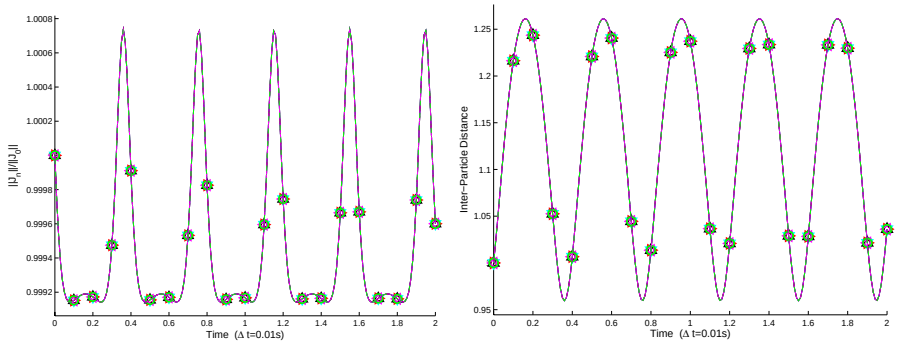
(f) Acceleration Norm

Figure 8.80: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option I) - U0V0(1.0,1.0,ρ_∞)]



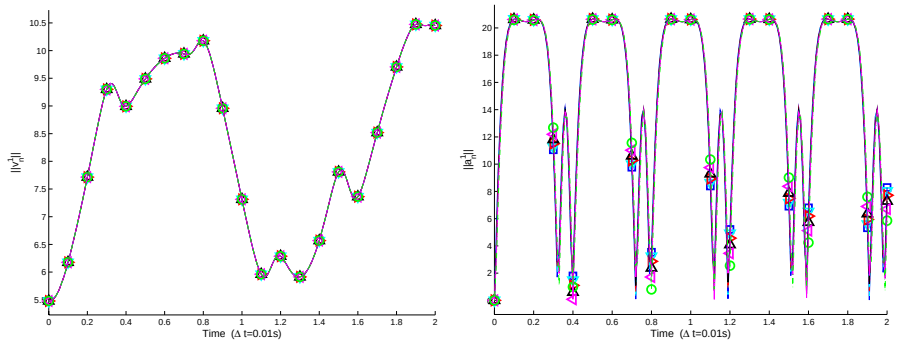
(a) Energy Ratio

(b) Linear Momentum



(c) Angular Momentum Ratio

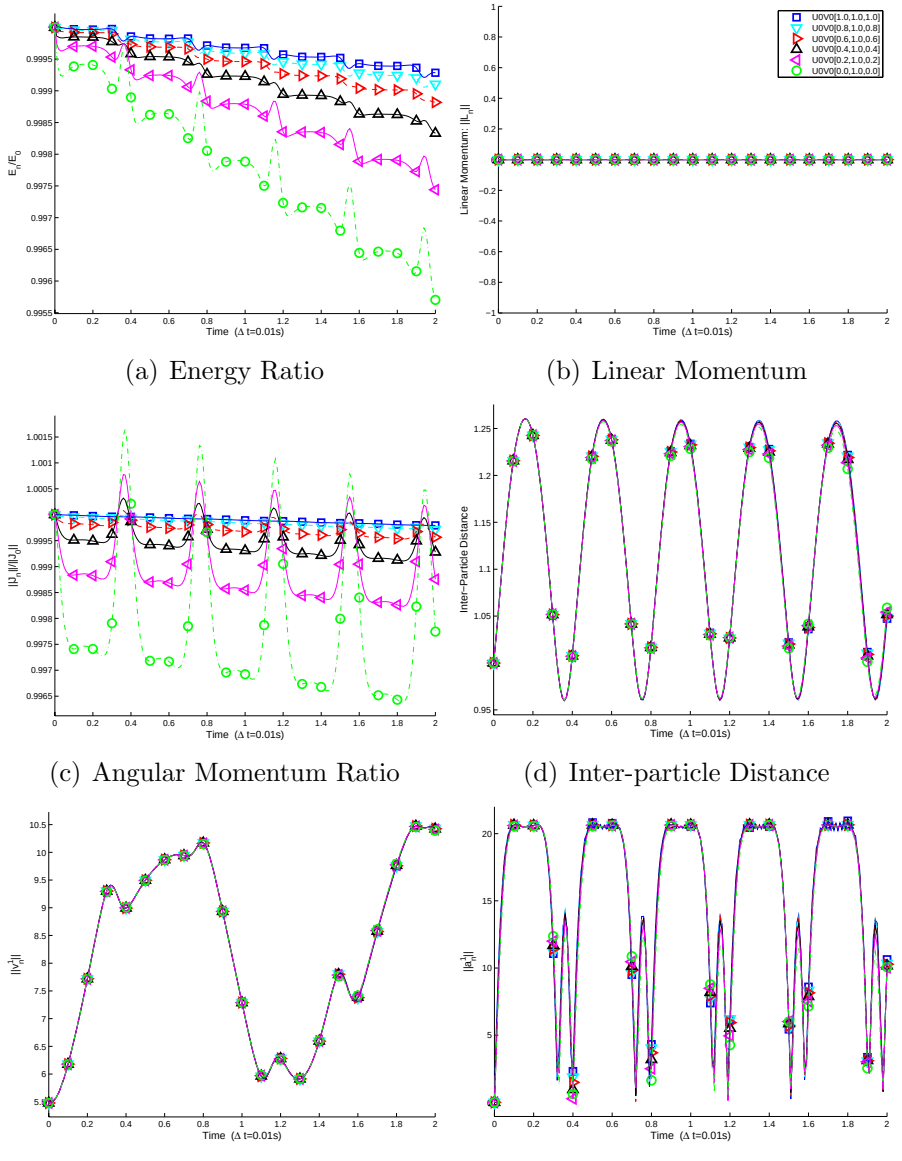
(d) Inter-particle Distance



(e) Velocity Norm

(f) Acceleration Norm

Figure 8.81: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option I) - V0U0(1.0,1.0,ρ_∞)]



(a) Energy Ratio

(b) Linear Momentum

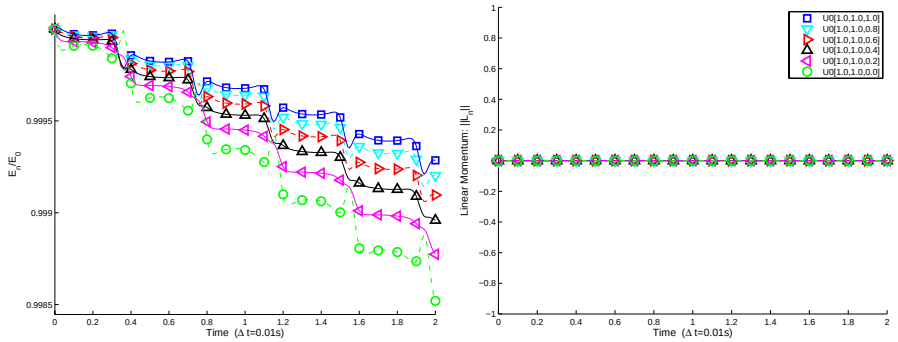
(c) Angular Momentum Ratio

(d) Inter-particle Distance

(e) Velocity Norm

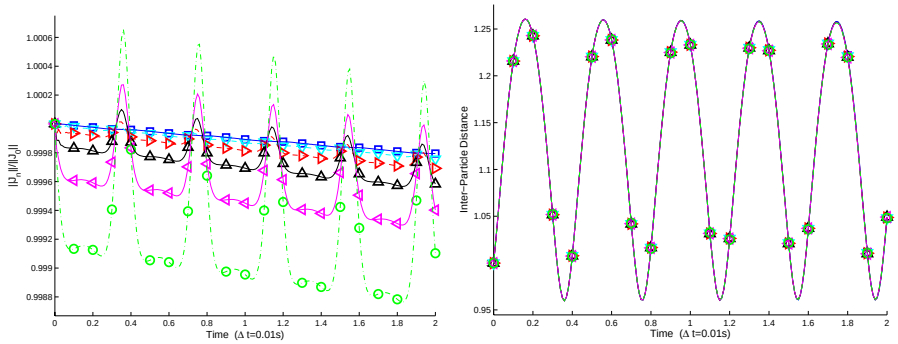
(f) Acceleration Norm

Figure 8.82: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option II) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



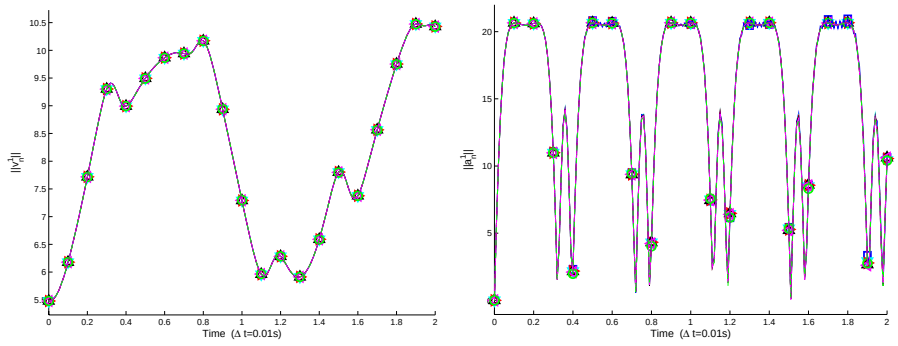
(a) Energy Ratio

(b) Linear Momentum



(c) Angular Momentum Ratio

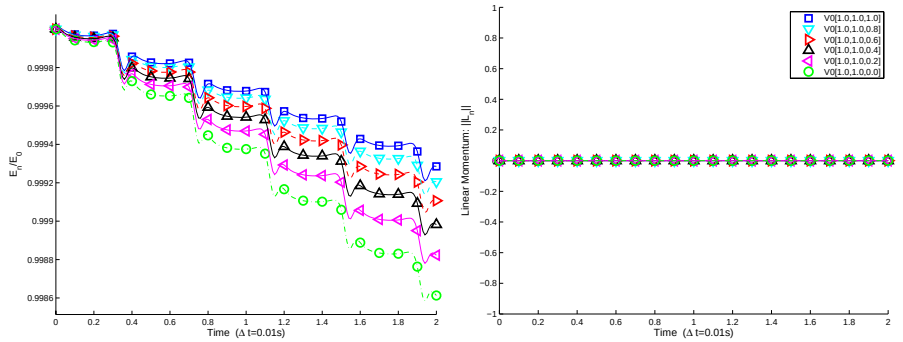
(d) Inter-particle Distance



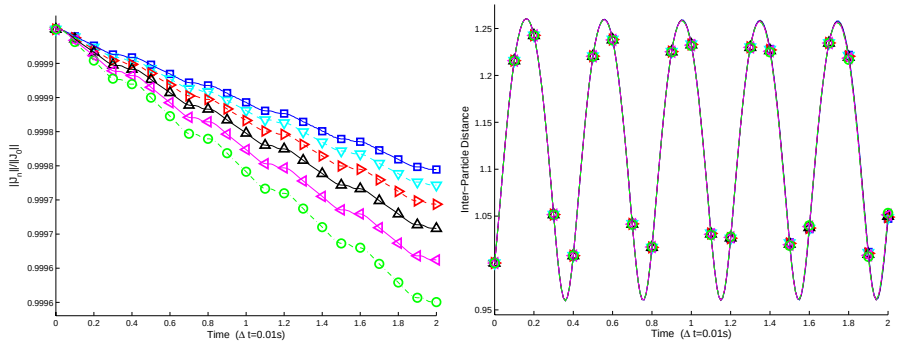
(e) Velocity Norm

(f) Acceleration Norm

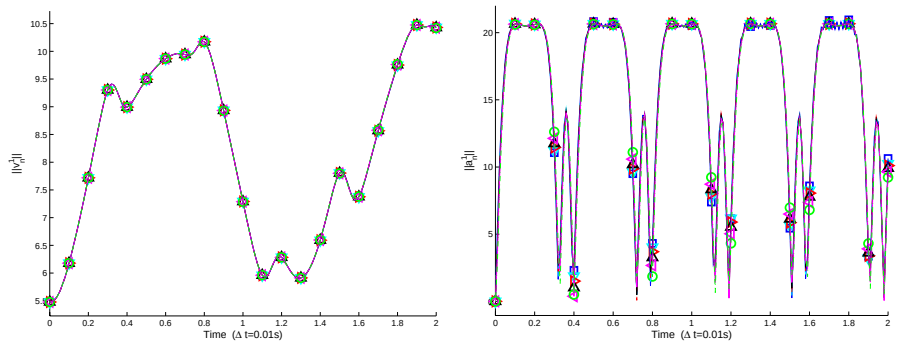
Figure 8.83: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option II) - U0V0(1.0,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum

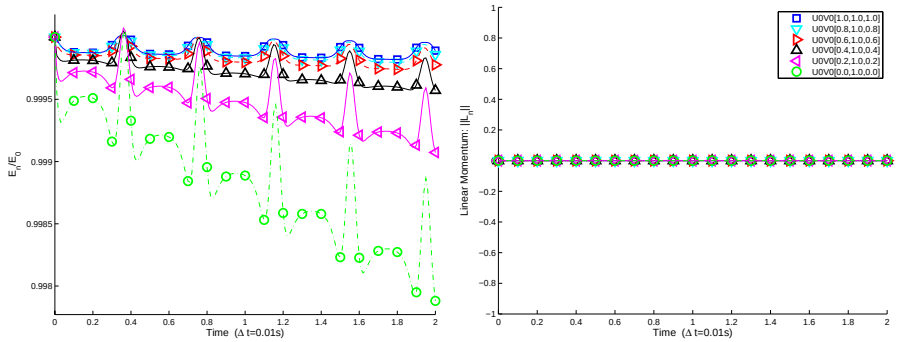


(c) Angular Momentum Ratio (d) Inter-particle Distance

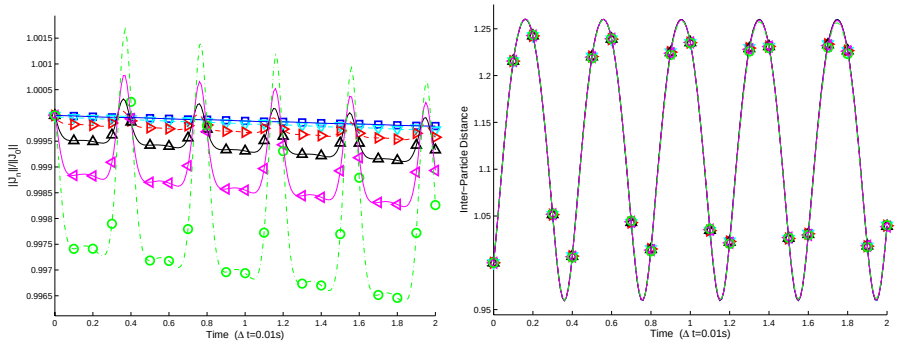


(e) Velocity Norm (f) Acceleration Norm

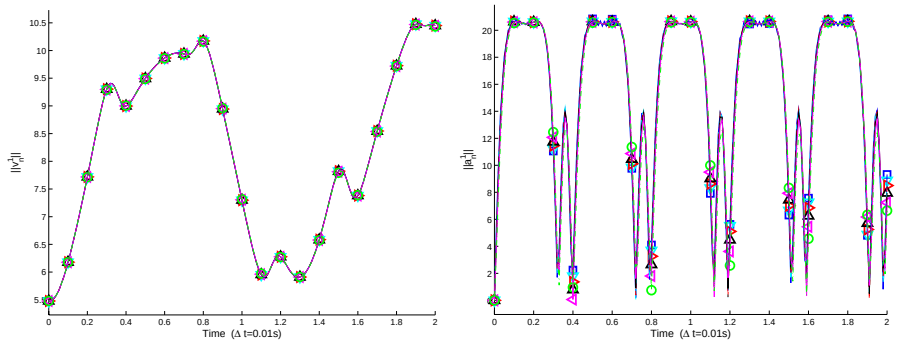
Figure 8.84: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option II) - V0U0(1.0,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum

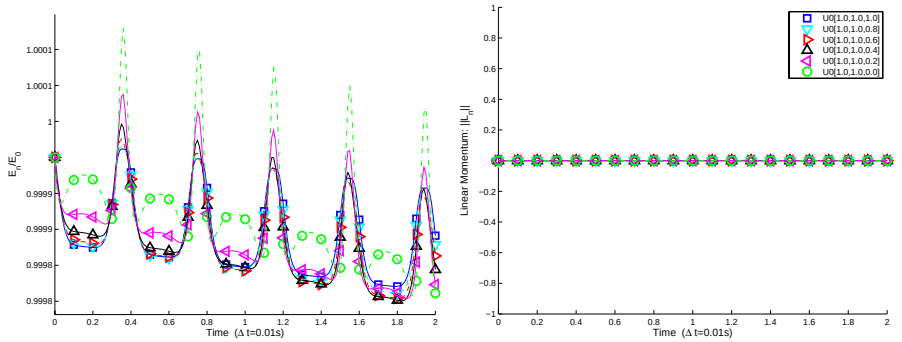


(c) Angular Momentum Ratio (d) Inter-particle Distance

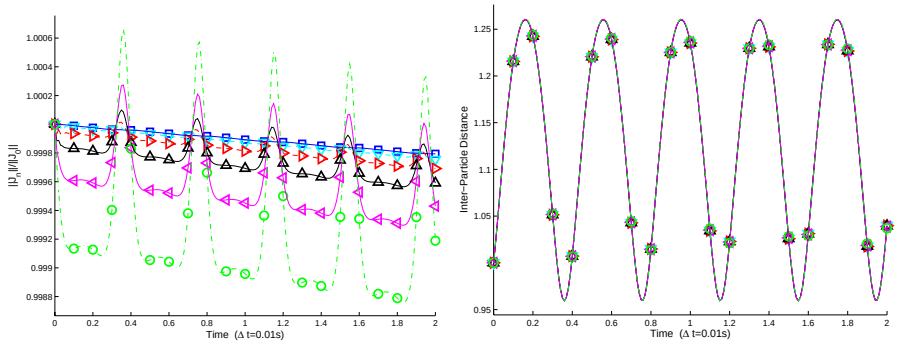


(e) Velocity Norm (f) Acceleration Norm

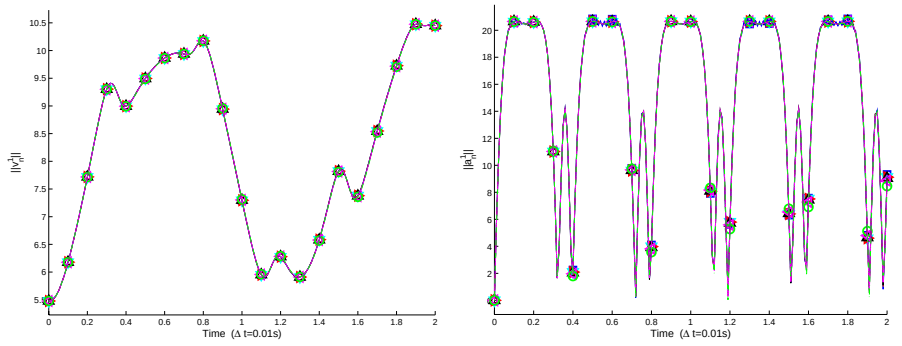
Figure 8.85: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option III) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum

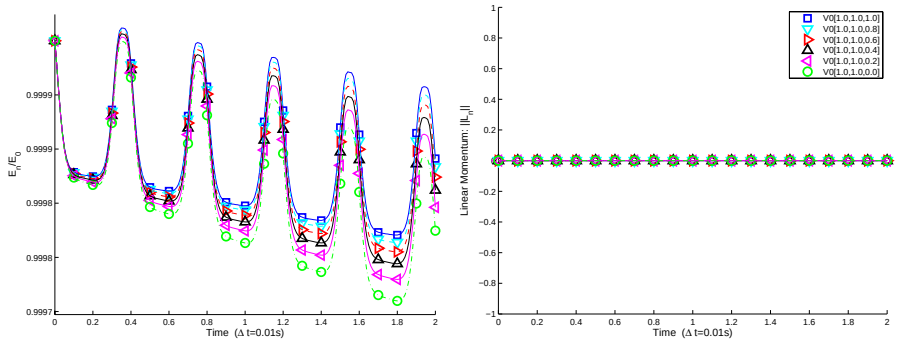


(c) Angular Momentum Ratio (d) Inter-particle Distance

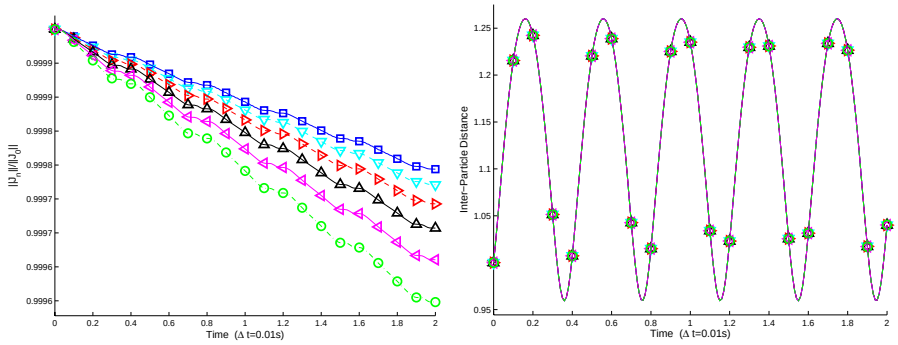


(e) Velocity Norm (f) Acceleration Norm

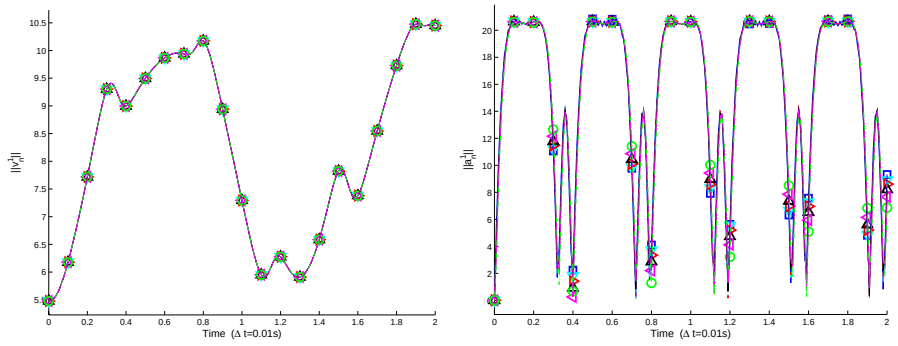
Figure 8.86: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option III) - U0V0(1.0,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum

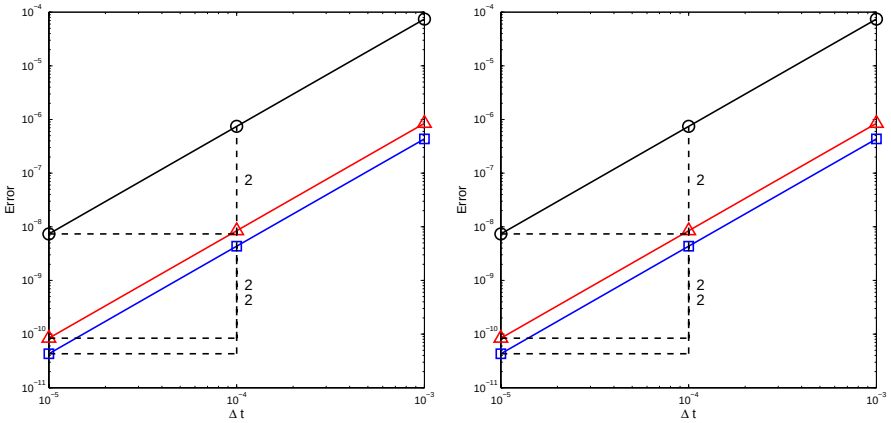


(c) Angular Momentum Ratio (d) Inter-particle Distance

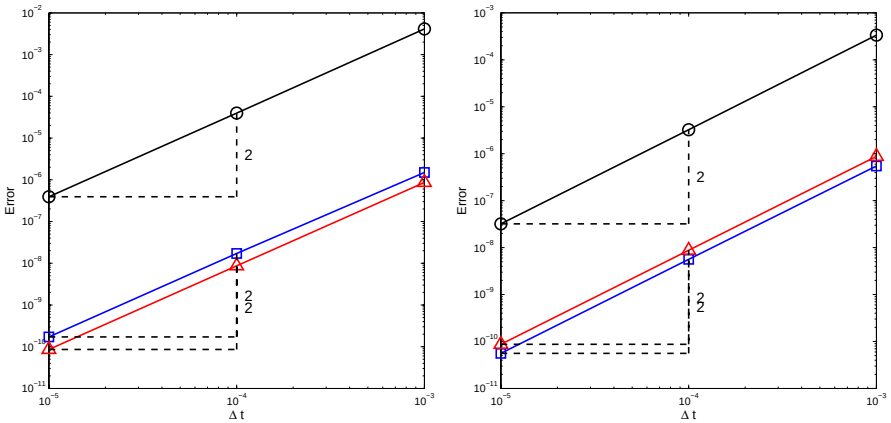


(e) Velocity Norm (f) Acceleration Norm

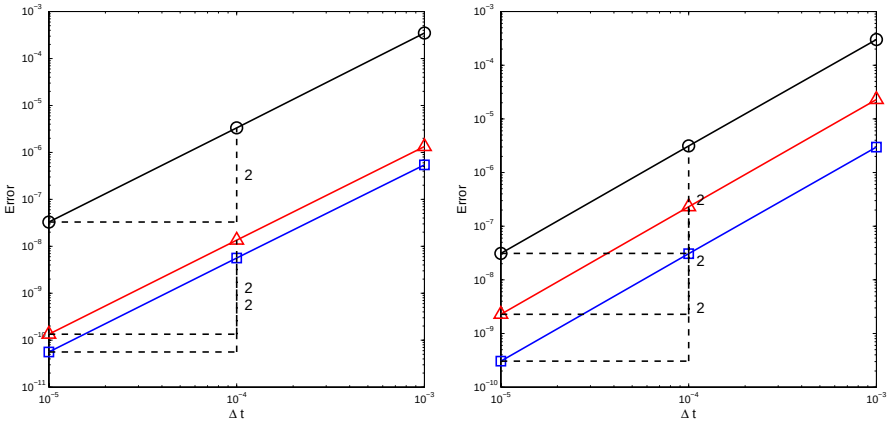
Figure 8.87: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 1$ (Option III) - V0U0(1.0,1.0,ρ_∞)]



(a) U0V0/V0U0(1.0,1.0,1.0) (b) U0V0(1.0,1.0,0.0)

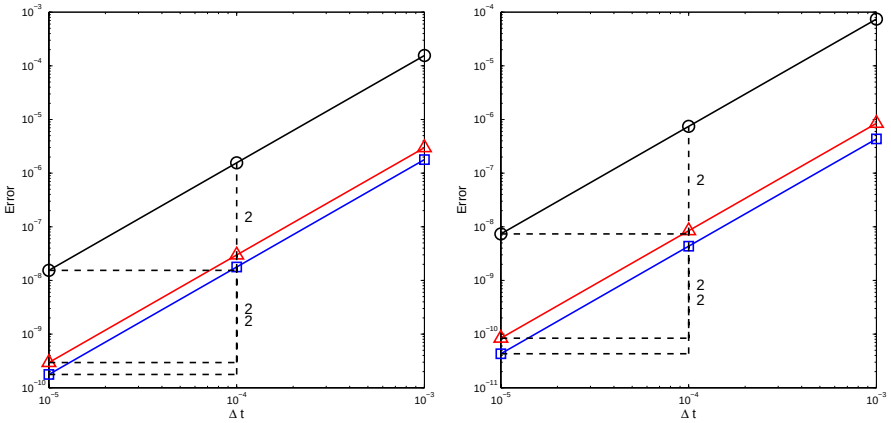


(c) V0U0(1.0,1.0,0.0) (d) U0V0/V0U0(0.8,1.0,0.8)



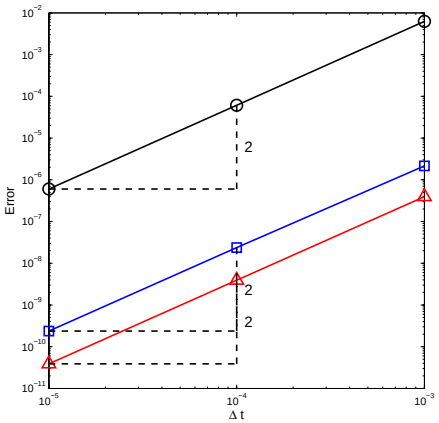
(e) U0V1(0.6,0.9,0.2) (f) V0U1(0.6,0.9,0.2)

Figure 8.88: Time accuracies in the configuration (□), velocity (△), and acceleration (○) in the forced mechanical system. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 0$ (Option I)]

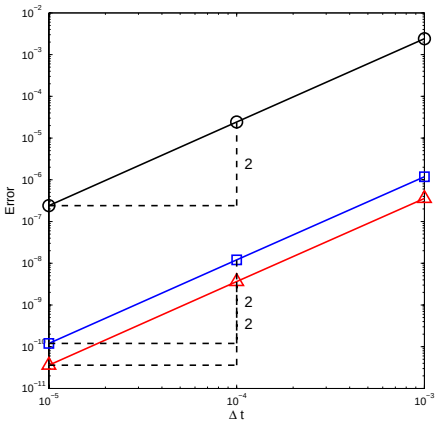


(a) U0V0/V0U0(1.0,1.0,1.0)

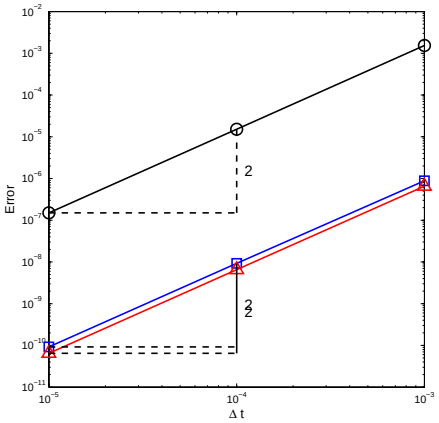
(b) U0V0(1.0,1.0,0.0)



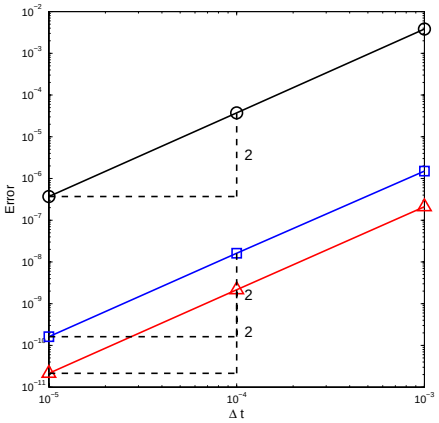
(c) V0U0(1.0,1.0,0.0)



(d) U0V0/V0U0(0.8,1.0,0.8)

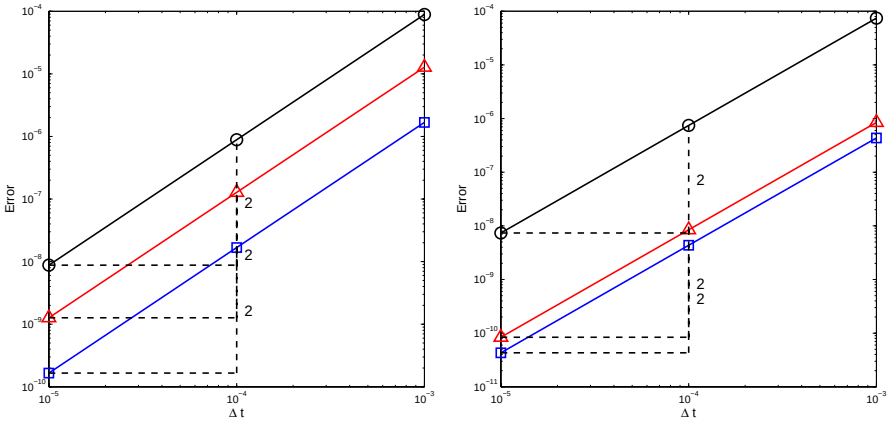


(e) U0V1(0.6,0.9,0.2)



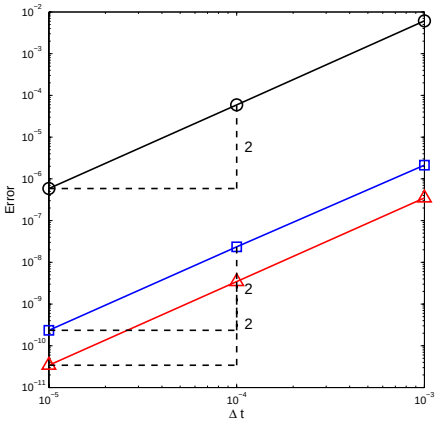
(f) V0U1(0.6,0.9,0.2)

Figure 8.89: Time accuracies in the configuration (□), velocity (△), and acceleration (○) in the forced mechani [Problem: Lennard-Jones potential problem] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 0$ (Option II)]

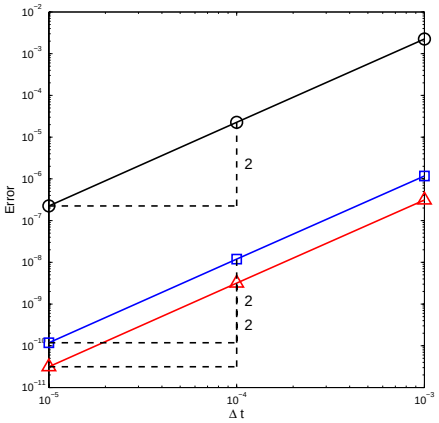


(a) U0V0/V0U0(1.0,1.0,1.0)

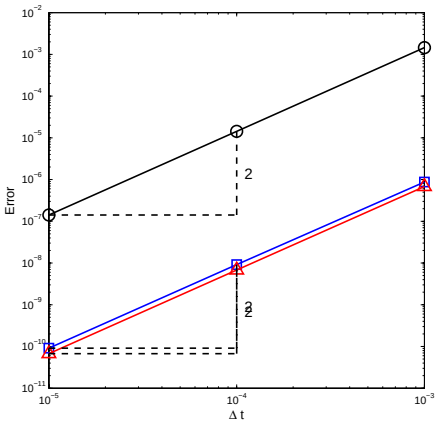
(b) U0V0(1.0,1.0,0.0)



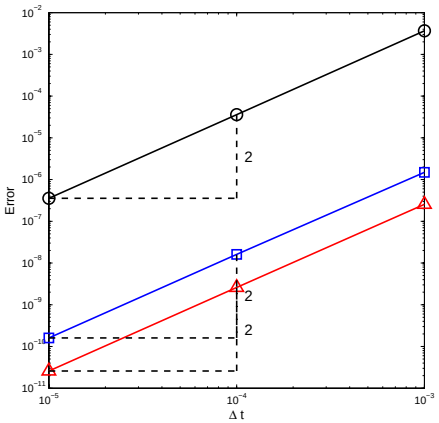
(c) V0U0(1.0,1.0,0.0)



(d) U0V0/V0U0(0.8,1.0,0.8)

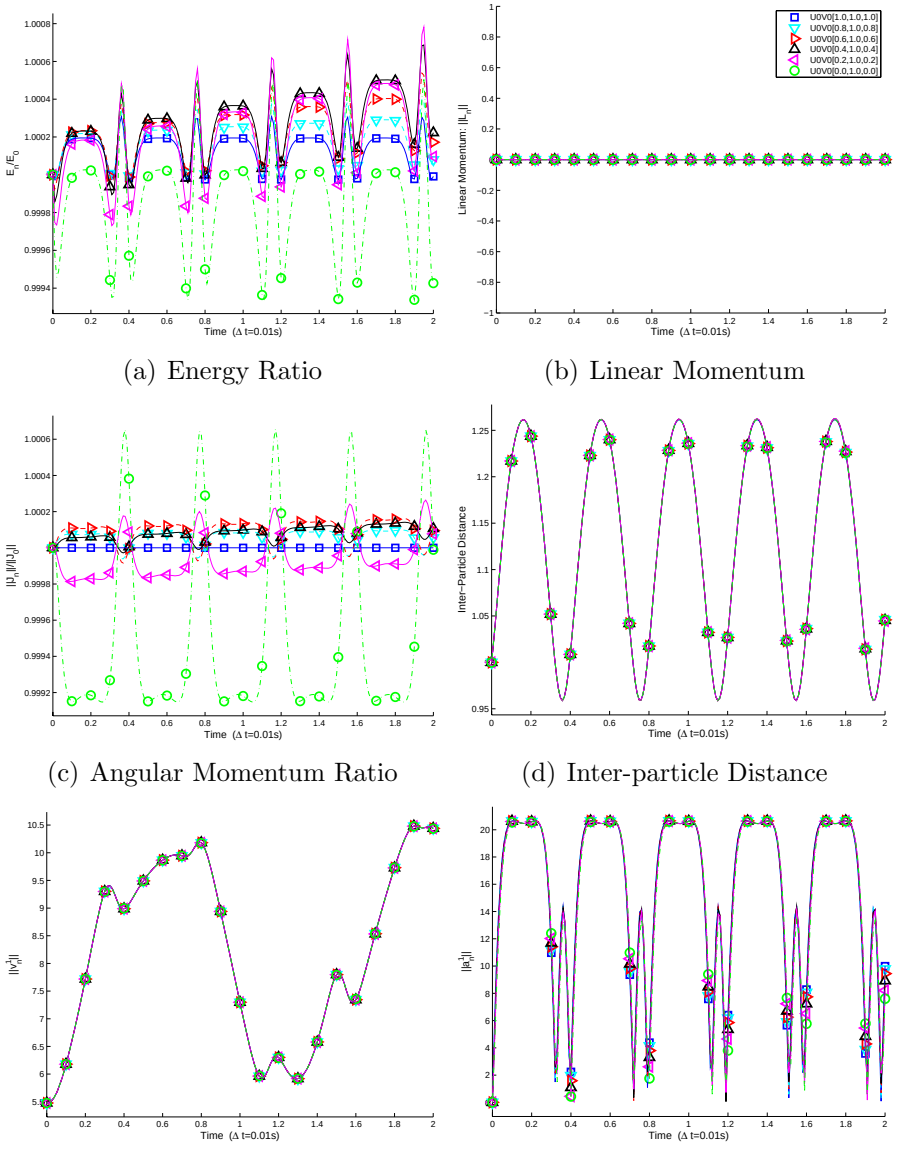


(e) U0V1(0.6,0.9,0.2)



(f) V0U1(0.6,0.9,0.2)

Figure 8.90: Time accuracies in the configuration, velocity, and acceleration in the forced mechanical system. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit (ET) GSSSS family of algorithms with $\eta_3 = 0$ (Option III)]



(a) Energy Ratio

(b) Linear Momentum

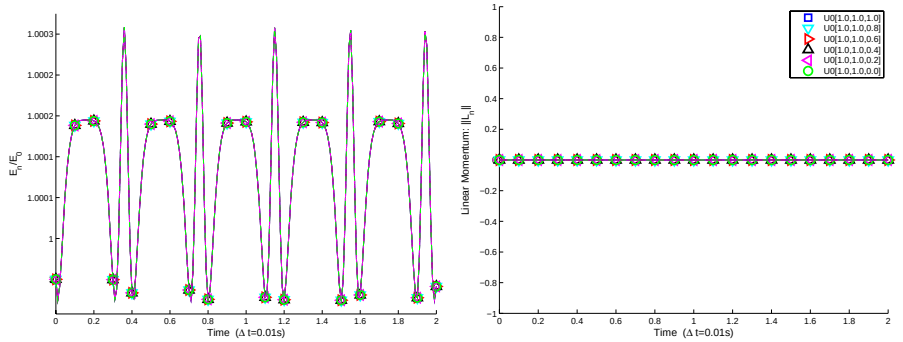
(c) Angular Momentum Ratio

(d) Inter-particle Distance

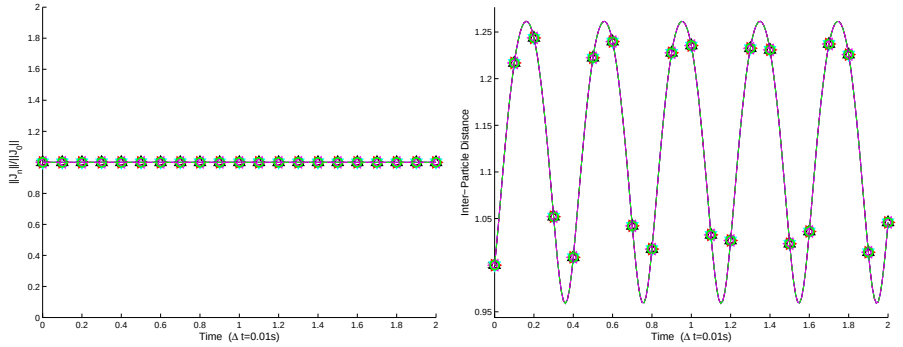
(e) Velocity Norm

(f) Acceleration Norm

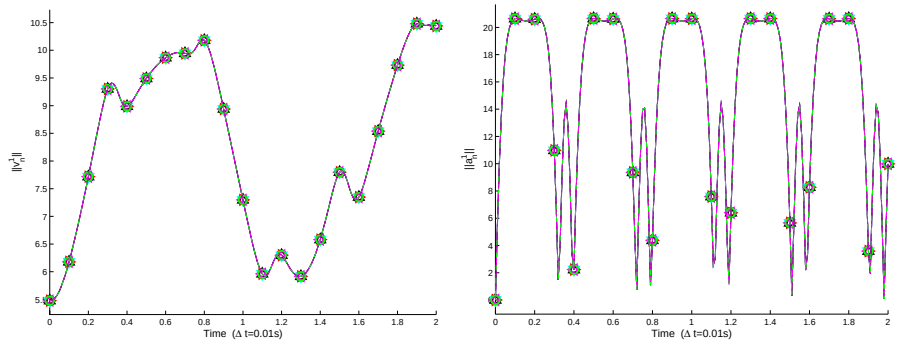
Figure 8.91: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option I) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum

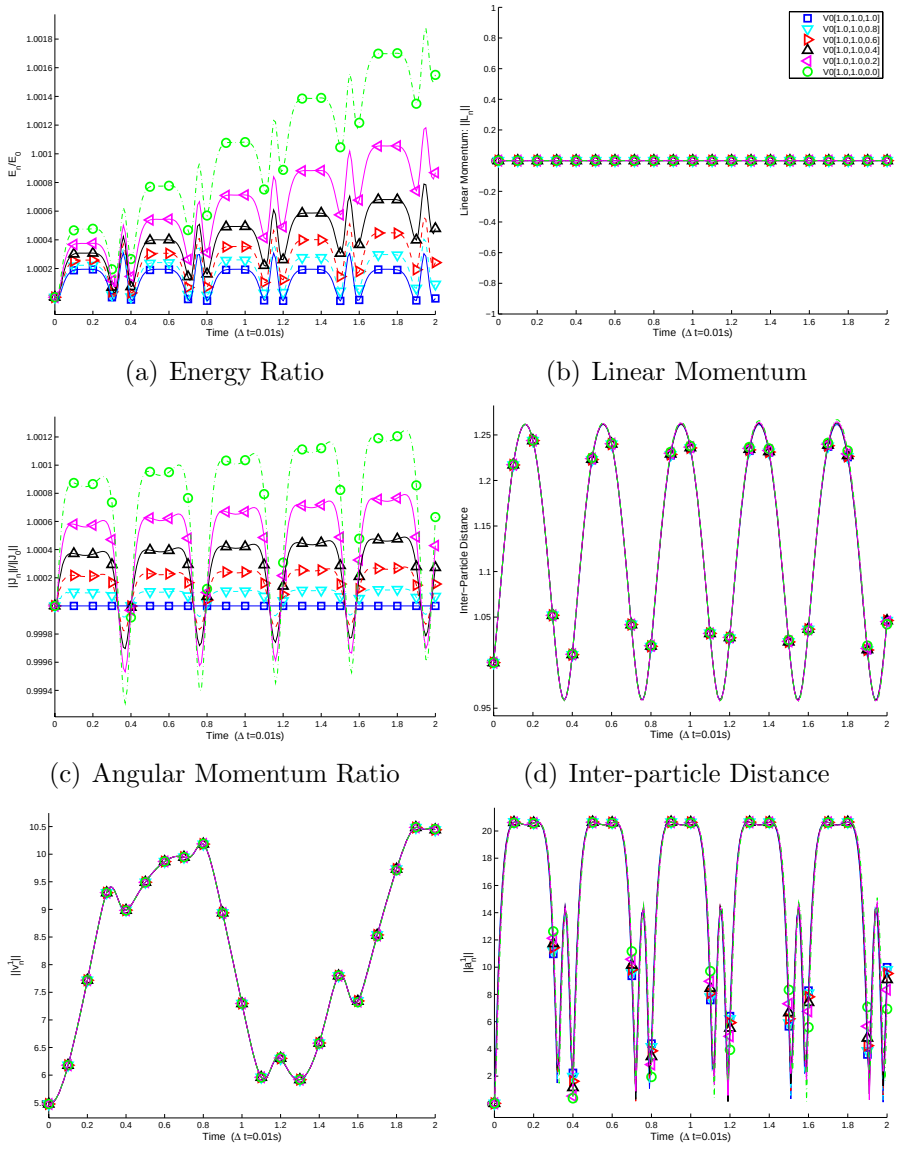


(c) Angular Momentum Ratio (d) Inter-particle Distance



(e) Velocity Norm (f) Acceleration Norm

Figure 8.92: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option I) - U0V0(1.0,1.0,ρ_∞)]



(a) Energy Ratio

(b) Linear Momentum

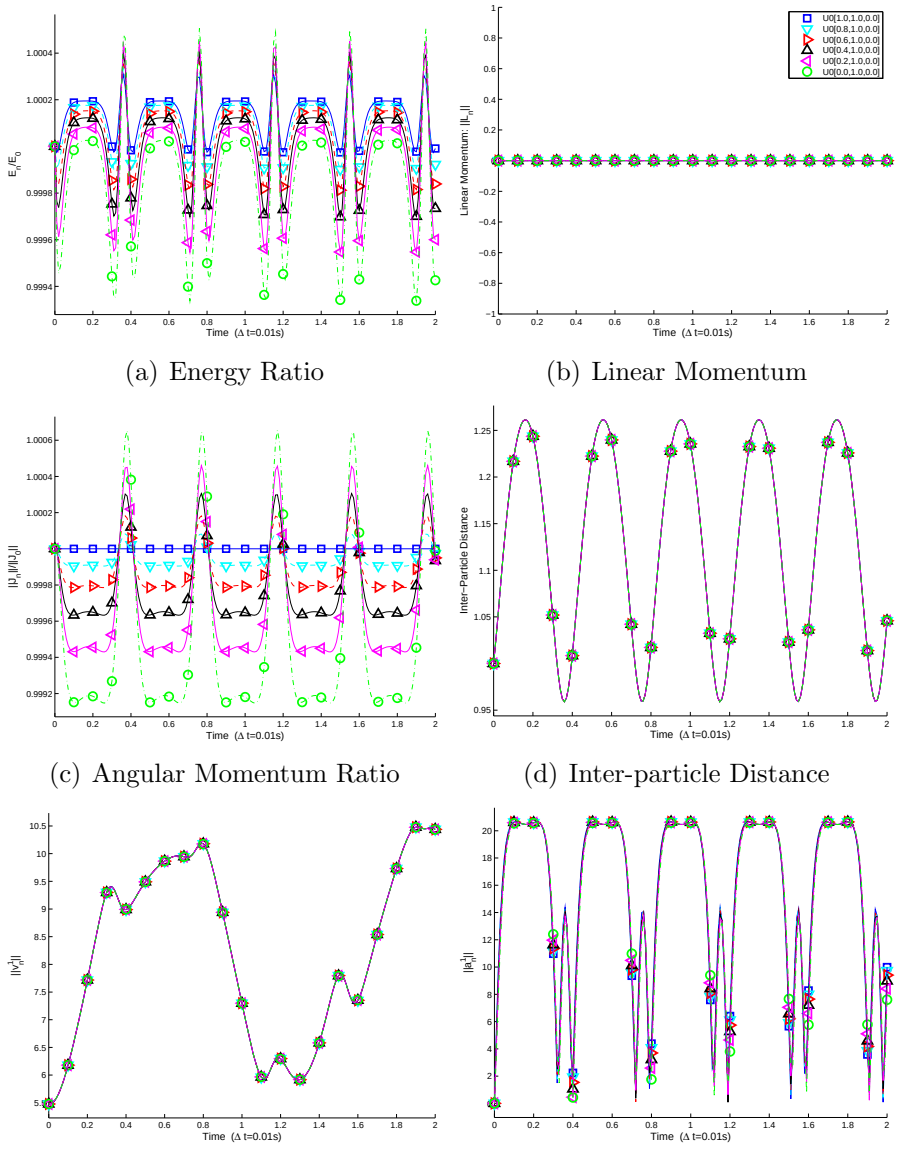
(c) Angular Momentum Ratio

(d) Inter-particle Distance

(e) Velocity Norm

(f) Acceleration Norm

Figure 8.93: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option I) - V0U0(1.0,1.0,ρ_∞)]



(a) Energy Ratio

(b) Linear Momentum

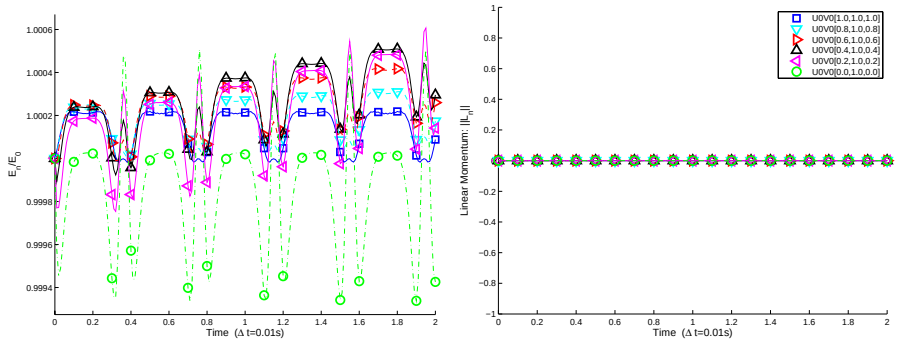
(c) Angular Momentum Ratio

(d) Inter-particle Distance

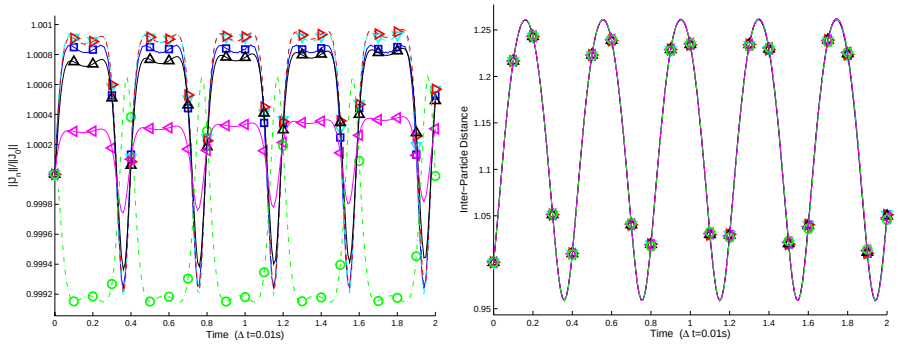
(e) Velocity Norm

(f) Acceleration Norm

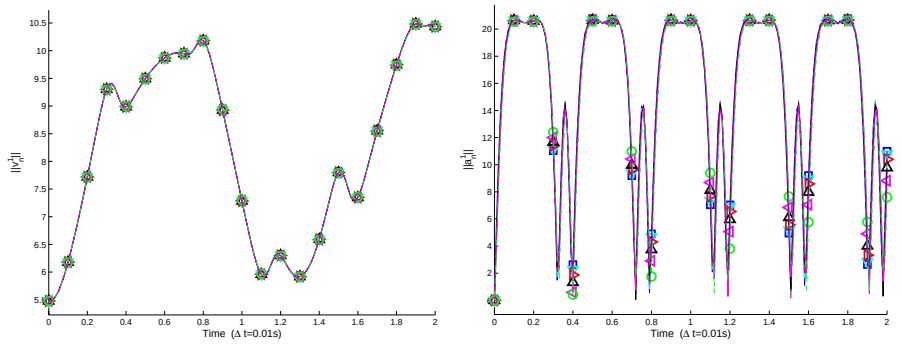
Figure 8.94: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option I) - U0(ρ_∞,1.0,0.0)]



(a) Energy Ratio (b) Linear Momentum

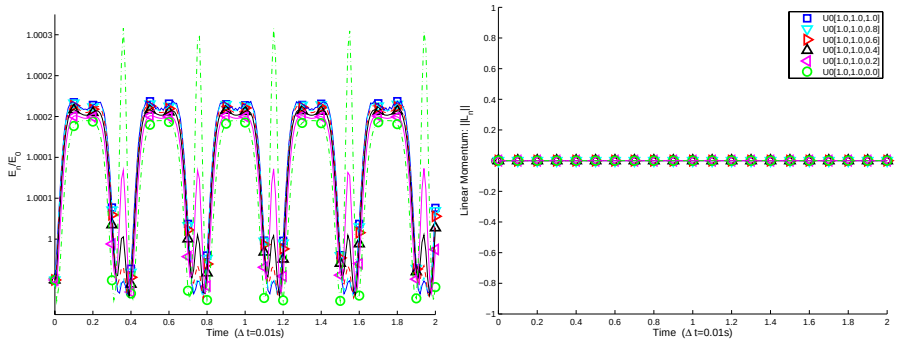


(c) Angular Momentum Ratio (d) Inter-particle Distance

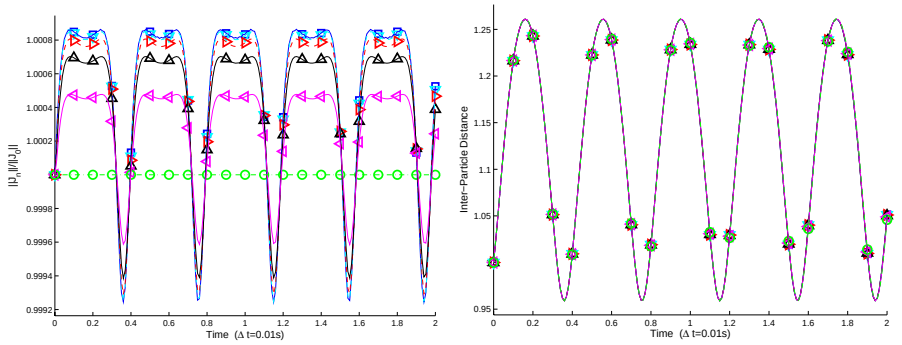


(e) Velocity Norm (f) Acceleration Norm

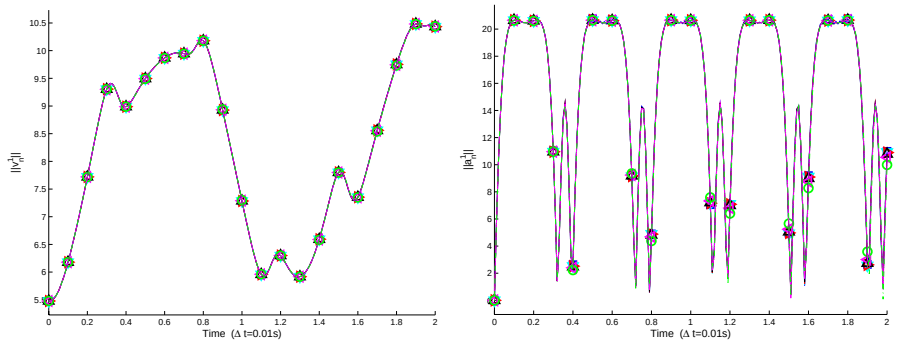
Figure 8.95: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option II) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum

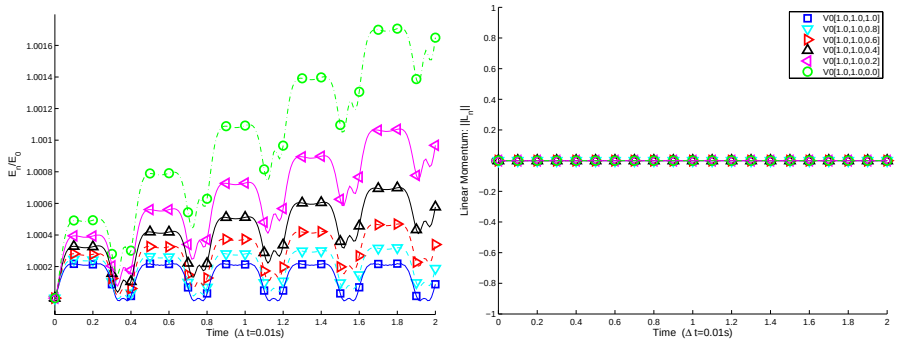


(c) Angular Momentum Ratio (d) Inter-particle Distance

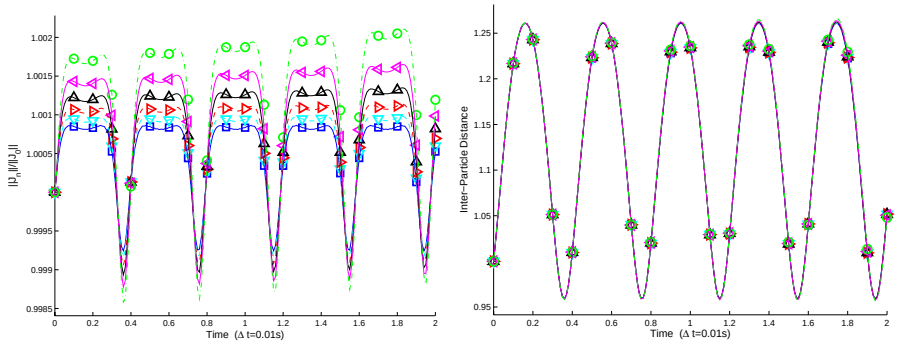


(e) Velocity Norm (f) Acceleration Norm

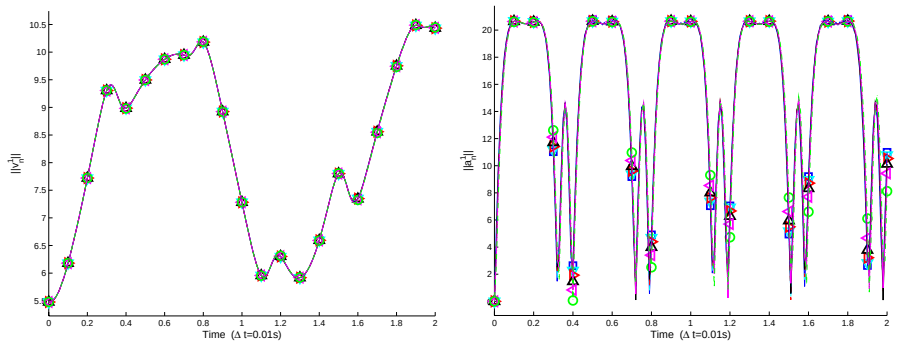
Figure 8.96: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option II) - U0V0(1.0,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum

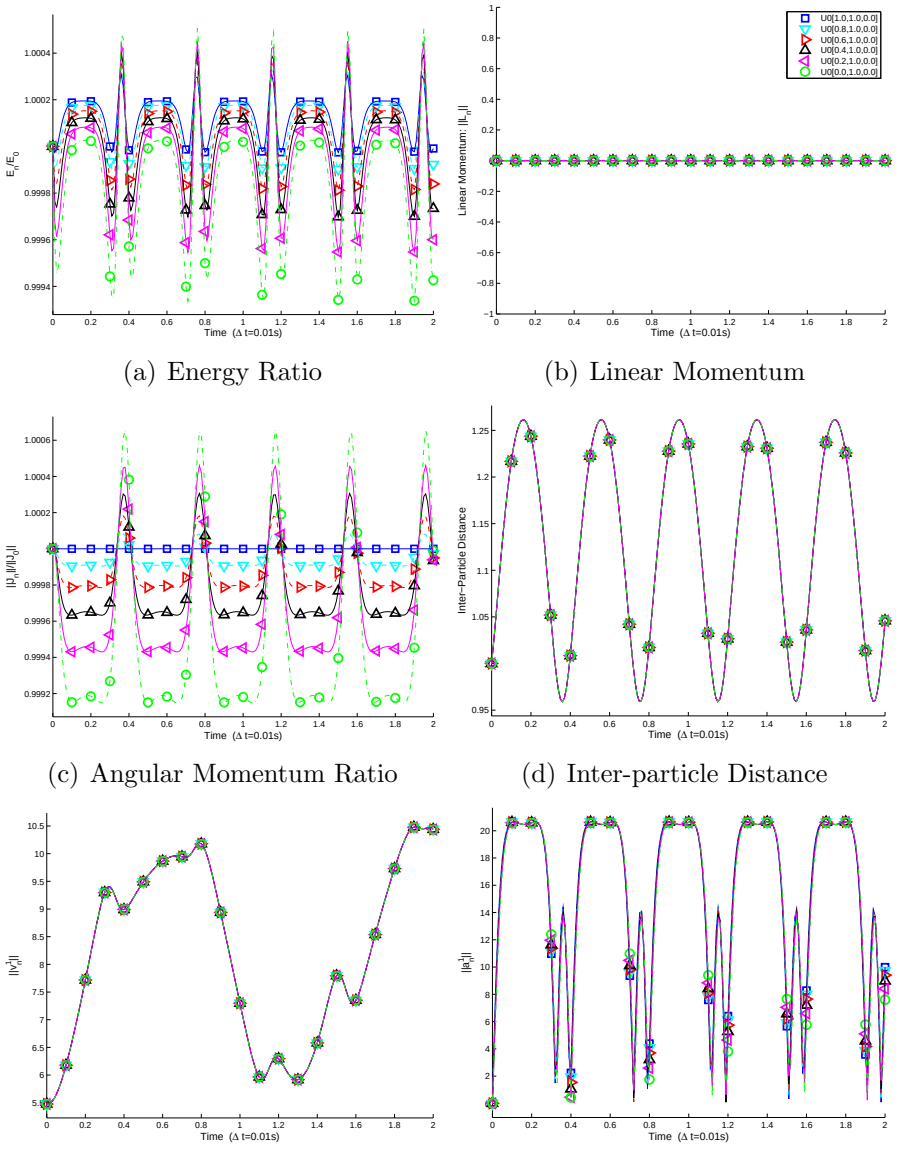


(c) Angular Momentum Ratio (d) Inter-particle Distance



(e) Velocity Norm (f) Acceleration Norm

Figure 8.97: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option II) - V0U0(1.0,1.0,ρ_∞)]



(a) Energy Ratio

(b) Linear Momentum

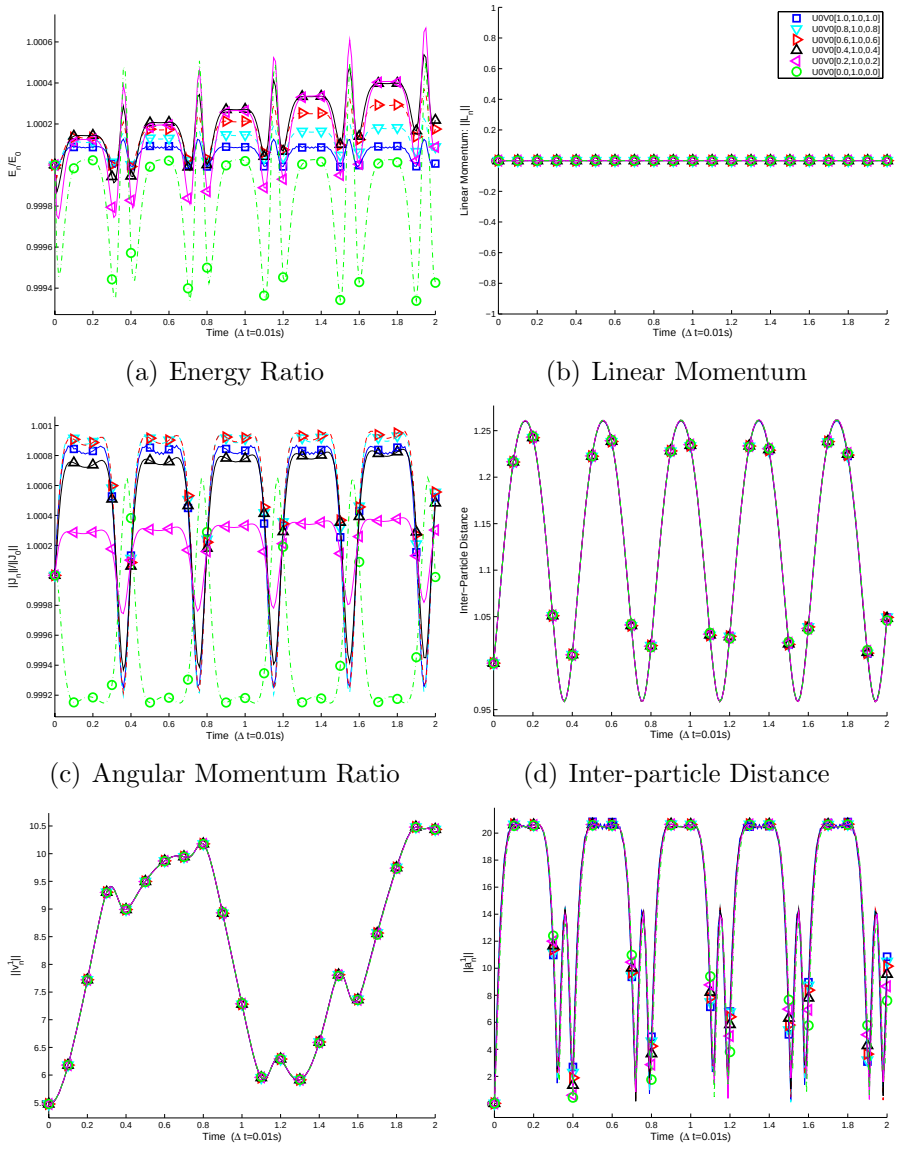
(c) Angular Momentum Ratio

(d) Inter-particle Distance

(e) Velocity Norm

(f) Acceleration Norm

Figure 8.98: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option II) - U0(ρ_∞,1.0,0.0)]



(a) Energy Ratio

(b) Linear Momentum

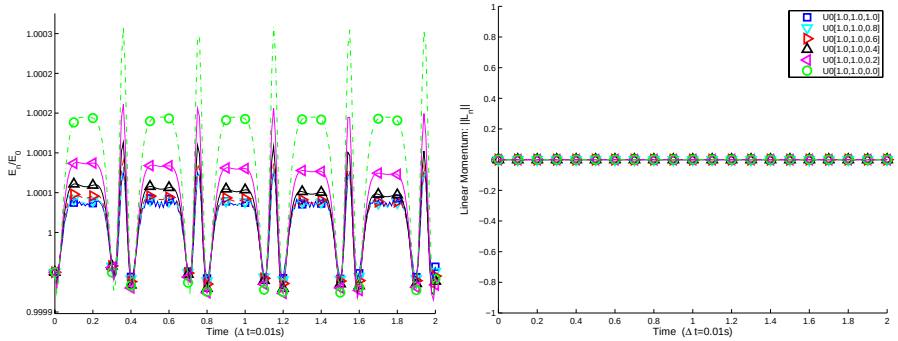
(c) Angular Momentum Ratio

(d) Inter-particle Distance

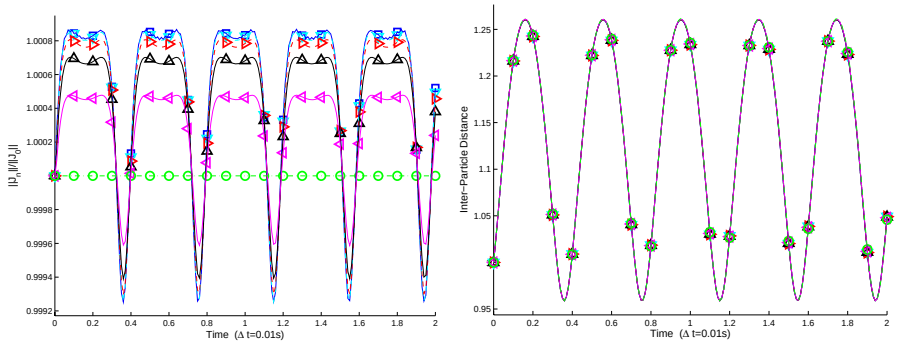
(e) Velocity Norm

(f) Acceleration Norm

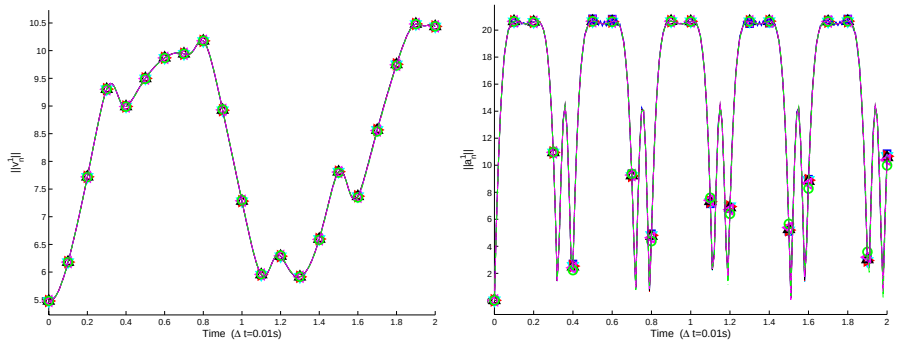
Figure 8.99: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option III) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum

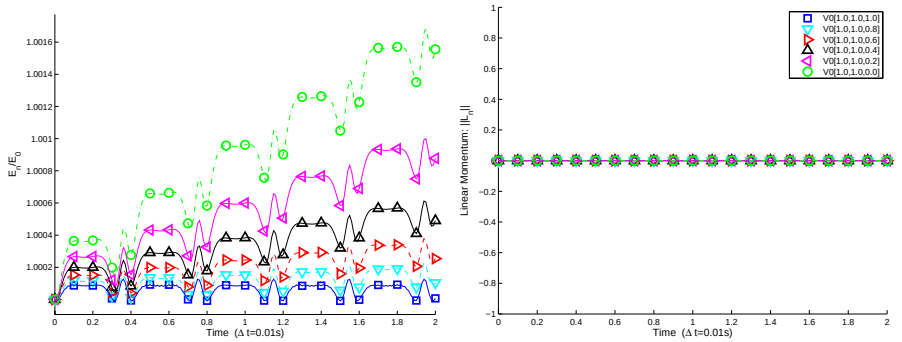


(c) Angular Momentum Ratio (d) Inter-particle Distance

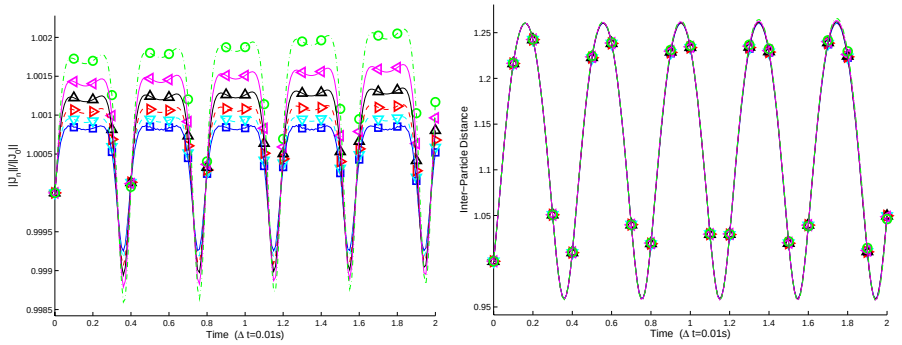


(e) Velocity Norm (f) Acceleration Norm

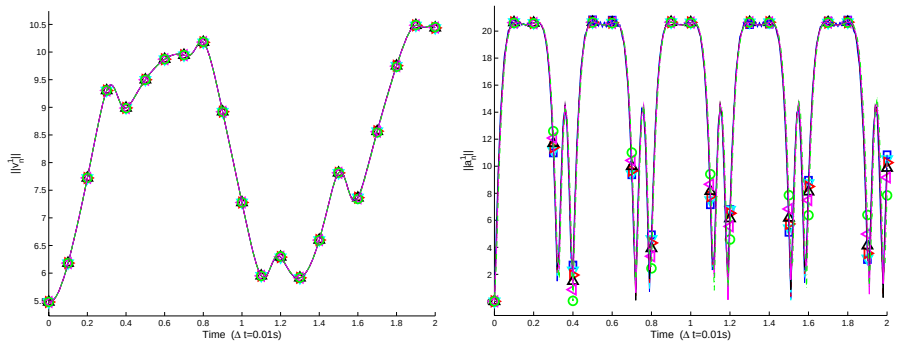
Figure 8.100: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option III) - U0V0(1.0,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum

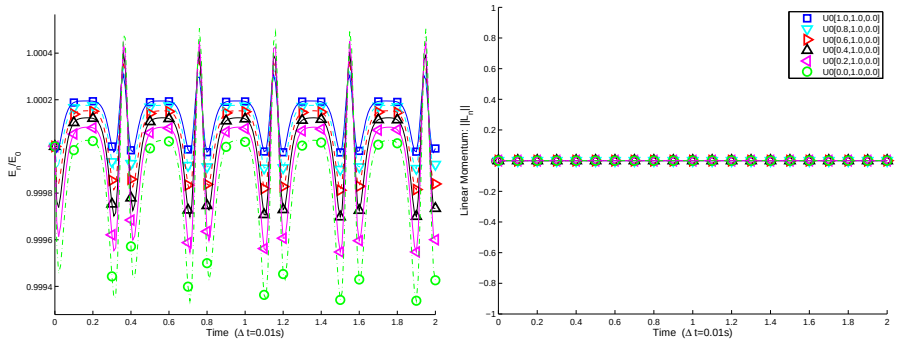


(c) Angular Momentum Ratio (d) Inter-particle Distance

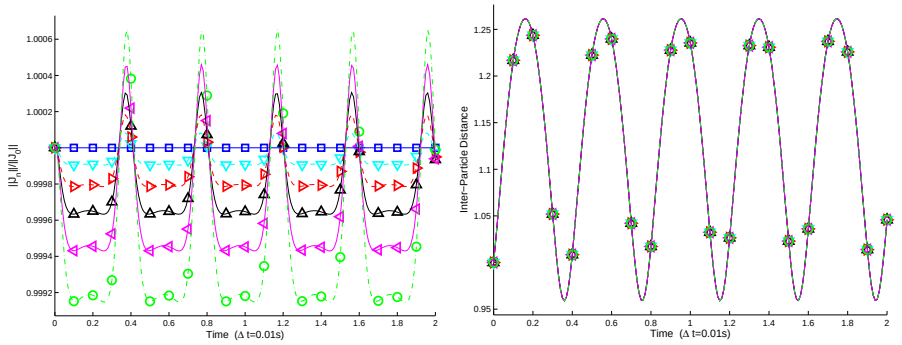


(e) Velocity Norm (f) Acceleration Norm

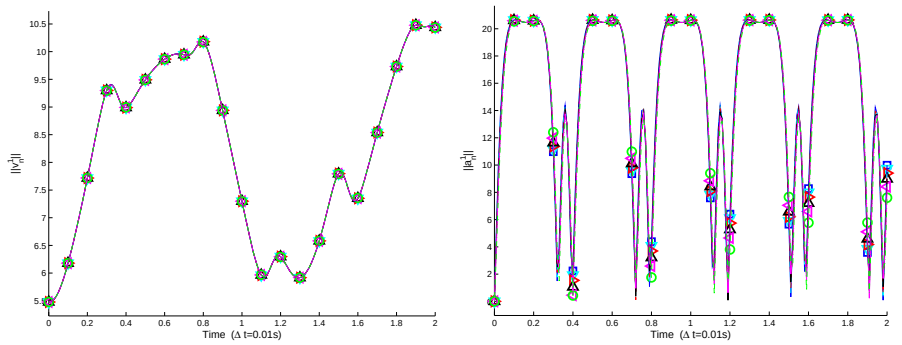
Figure 8.101: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option III) - V0U0(1.0,1.0,ρ_∞)]



(a) Energy Ratio (b) Linear Momentum

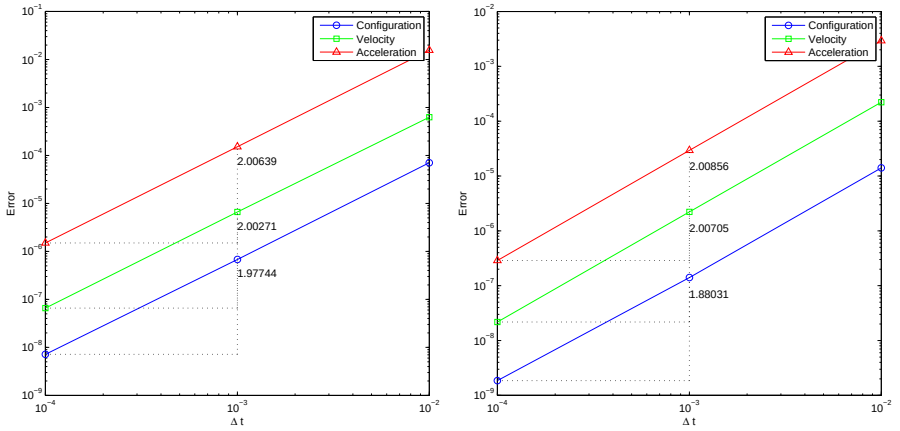


(c) Angular Momentum Ratio (d) Inter-particle Distance



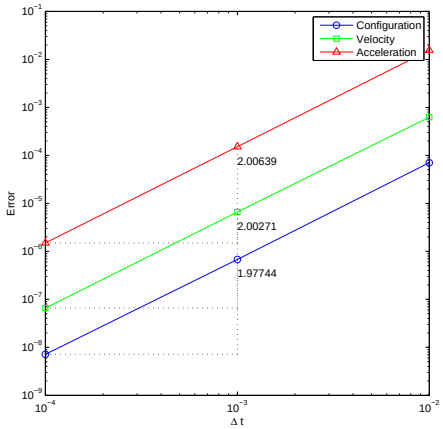
(e) Velocity Norm (f) Acceleration Norm

Figure 8.102: Time histories in the ***conservative system***. [Problem: Lennard-Jones potential problem] [Algorithm: Explicit GSSSS family of algorithms with $\eta_3 = 0$ (Option III) - U0(ρ_∞,1.0,0.0)]

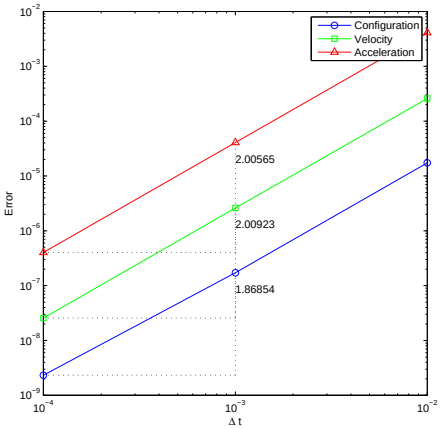


(a) U0V0/V0U0(1.0,1.0,1.0)

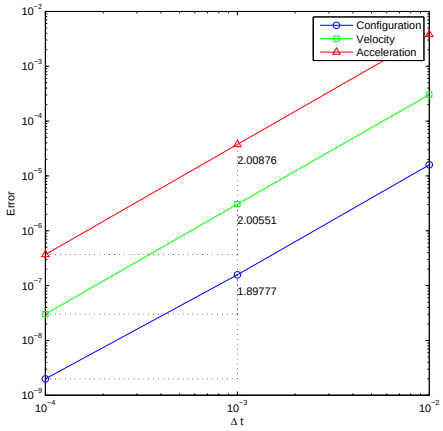
(b) U0V0(1.0,1.0,0.0)



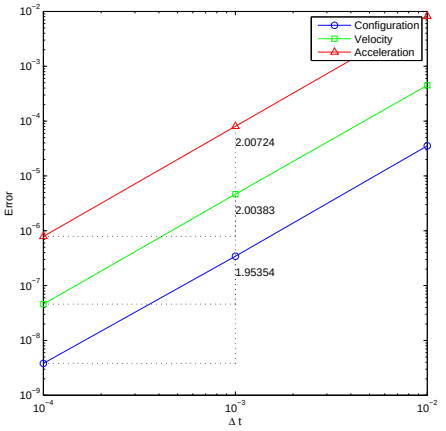
(c) V0U0(1.0,1.0,0.0)



(d) U0V0/V0U0(0.8,1.0,0.8)

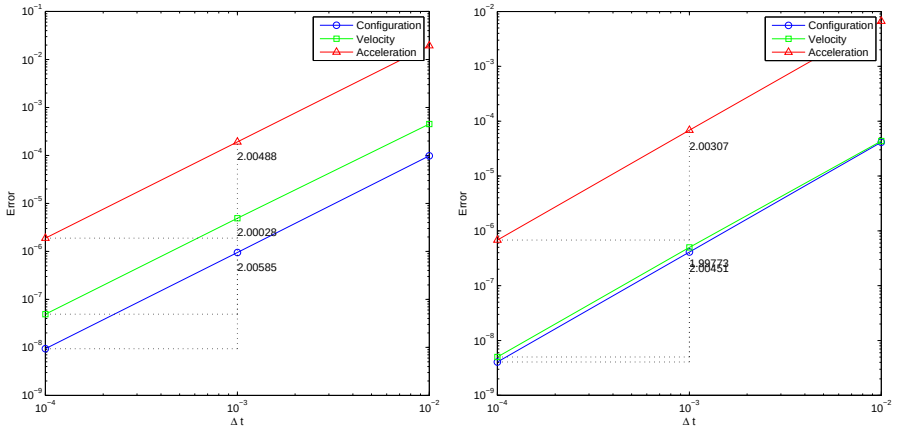


(e) U0V1(0.6,0.9,0.2)



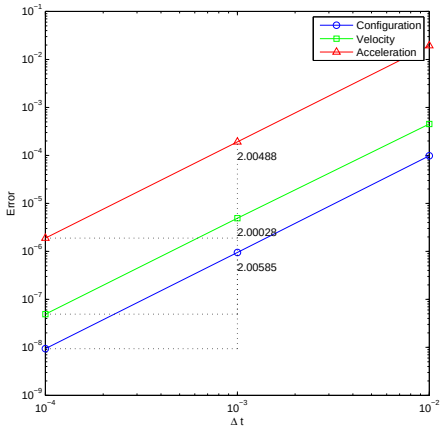
(f) V0U1(0.6,0.9,0.2)

Figure 8.103: Time accuracies in the configuration (□), velocity (△), and acceleration (○) in the forced mechanical system. [Problem: Spring-Pendulum Problem] [Algorithm: PCE-IT GSSSS family of algorithms with $\eta_3 = 0$ (Option II)]

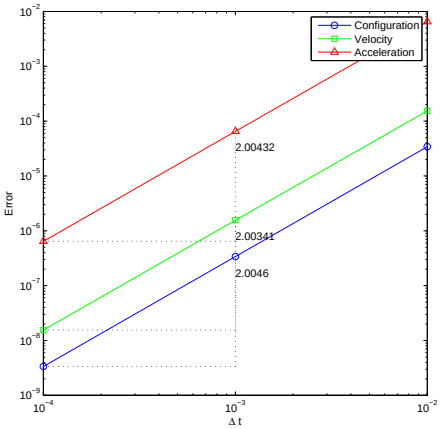


(a) U0V0/V0U0(1.0,1.0,1.0)

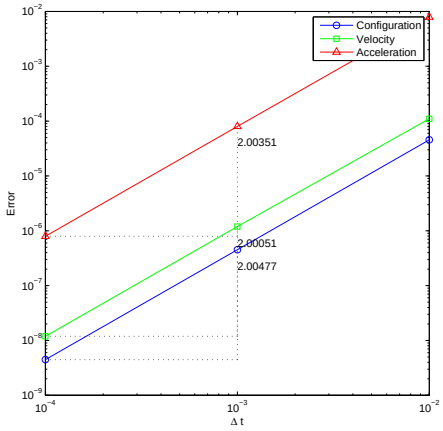
(b) U0V0(1.0,1.0,0.0)



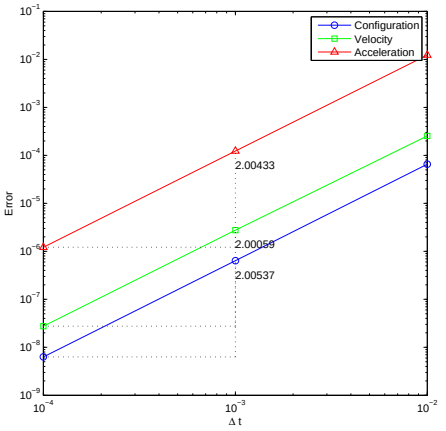
(c) V0U0(1.0,1.0,0.0)



(d) U0V0/V0U0(0.8,1.0,0.8)



(e) U0V1(0.6,0.9,0.2)



(f) V0U1(0.6,0.9,0.2)

Figure 8.104: Time accuracies in the configuration (□), velocity (△), and acceleration (○) in the forced mechanical system. [Problem: Spring-Pendulum Problem] [Algorithm: PCE-ET GSSSS family of algorithms with $\eta_3 = 0$ (Option II)]

Chapter 9

A Novel *i*INTEGRATION Framework and Architecture: Isochronous Time Discretization of Balance Equations in the Second- and First-Order Systems

In this chapter, we introduce a novel architecture and a unified implicit and explicit time integration framework which can be applied not only to second-order systems, but simultaneously also to first-order systems. Strictly speaking, although we show the general designs of the second- and first-order time systems separately as GSSSS-2 and GSSSS-1 frameworks, only the former is sufficient for adapting to both systems. Hence, the term ***isochronous*** is used since the same framework is ideal for either or both systems. The initial-value problem in the single-field form for nonlinear dynamical systems consists of the second-order balance equation (equation of motion) in the single-field form and the given initial conditions. That

is,

$$\boxed{\begin{aligned}&\textbf{\textit{Balance Equation:}}\\&\mathbf{M}\ddot{\mathbf{q}}(t) = \mathbf{f}^{\text{appl}}(\mathbf{q}, \dot{\mathbf{q}}, t) \quad \forall t \in \mathbb{I}\\&\textbf{\textit{Initial conditions (Given):}}\\&\mathbf{q}(t_0) = \mathbf{q}_0\\&\dot{\mathbf{q}}(t_0) = \dot{\mathbf{q}}_0\end{aligned}} \tag{9.1}$$

where $\mathbf{M} \in \mathbb{R}^{n_{\text{dof}} \times n_{\text{dof}}}$ and $\mathbf{f}^{\text{appl}}(\mathbf{q}, \dot{\mathbf{q}}, t) : TQ \times \mathbb{I} \to \mathbb{R}^{n_{\text{dof}}}$ are the mass matrix and the total resultant force vector, respectively; and $\mathbf{q}(t) : \mathbb{I} \to Q \equiv \mathbb{R}^{n_{\text{dof}}}$, $\dot{\mathbf{q}}(t) : \mathbb{I} \to T_{\mathbf{q}}Q \equiv \mathbb{R}^{n_{\text{dof}}}$, $\ddot{\mathbf{q}}(t) : \mathbb{I} \to \mathbb{R}^{n_{\text{dof}}}$ are the configuration, velocity, and acceleration vectors, respectively, where n_{dof} denotes the number of degrees of freedom. The balance equation above is defined in the whole time interval $\mathbb{I} := [t_0 = 0, T] \subset \mathbb{R}_+$. On the other hand, the initial-value problem for first-order parabolic or hyperbolic systems, such as heat transfer problems, fluid flow problems, electromagnetic problems, etc., consists of

$$\boxed{\begin{aligned}&\textbf{\textit{Balance Equation:}}\\&\mathbf{M}\dot{\mathbf{q}}(t) = \mathbf{f}^{\text{appl}}(\mathbf{q}, t) \quad \forall t \in \mathbb{I}\\&\textbf{\textit{Initial condition (Given):}}\\&\mathbf{q}(t_0) = \mathbf{q}_0\end{aligned}} \tag{9.2}$$

where $\mathbf{M} \in \mathbb{R}^{n_{\text{dof}} \times n_{\text{dof}}}$ and $\mathbf{f}^{\text{appl}}(\mathbf{q}, t) : Q \times \mathbb{I} \to \mathbb{R}^{n_{\text{dof}}}$ are the capacity matrix and the total resultant supply vector, respectively; $\mathbf{q}(t) : \mathbb{I} \to Q \equiv \mathbb{R}^{n_{\text{dof}}}$ and $\dot{\mathbf{q}}(t) : \mathbb{I} \to T_{\mathbf{q}}Q \equiv \mathbb{R}^{n_{\text{dof}}}$ are the nodal temperature vector and the time derivative of the nodal temperature vector, respectively. In both Eq. (9.1) and Eq. (9.2), $\mathbf{M}$ is symmetric and positive-definite.

9.1 Implicit and Explicit Algorithms by Design in Second- and First-Order Systems: Generalized Single Step Single Solve Algorithms, Designs, and Framework

We introduce a partition of $\mathbb{I}$ by $0 \leq t_0 < t_1 < t_2 < \cdots < t_n < \cdots < t_f \equiv T$, and define the time step intervals, $\Delta t := t_{n+1} - t_n$ for $n = 0, 1, 2, \cdots, f-1$.

Second-order Systems

For the general case, the time discretization of Eq. (9.1) via the ***normalized time weighted residual methodology*** leads to the so-called implicit generalized single step single solve algorithms (***I-GSSSS-2 algorithms***) which comprises of the zero-order configuration overshooting schemes (***U0 family***) and the zero-order velocity overshooting schemes (***V0 family***) as the basis. The summary of the framework of algorithms is given below.

Algorithm 31

Framework of Implicit and Explicit GSSSS Family of Algorithms and Designs for Nonlinear Dynamical Systems in Second-order Systems Integrator:

$$\mathbf{M}\tilde{\mathbf{a}} = \mathbf{f}^{\text{appl}}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}, t_{n+W_1})$$

where

$$\begin{aligned}
\tilde{\mathbf{a}} &= \ddot{\mathbf{q}}_n + W_1\Lambda_6\Delta\mathbf{a} \\
\tilde{\mathbf{v}} &= \dot{\mathbf{q}}_n + W_1\Lambda_4\ddot{\mathbf{q}}_n\Delta t + W_2\Lambda_5\Delta\mathbf{a}\Delta t\eta_1 \\
\tilde{\mathbf{q}} &= \mathbf{q}_n + W_1\Lambda_1\dot{\mathbf{q}}_n\Delta t + W_2\Lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + W_3\Lambda_3\Delta\mathbf{a}\Delta t^2\eta_2
\end{aligned}$$

Updates:

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \lambda_1 \dot{\mathbf{q}}_n \Delta t + \lambda_2 \ddot{\mathbf{q}}_n \Delta t^2 + \lambda_3 \Delta \mathbf{a} \Delta t^2 \eta_3$$
$$\dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_n + \lambda_4 \ddot{\mathbf{q}}_n \Delta t + \lambda_5 \Delta \mathbf{a} \Delta t$$
$$\ddot{\mathbf{q}}_{n+1} = \ddot{\mathbf{q}}_n + \Delta \mathbf{a}$$

Initial conditions:

$$\mathbf{q}(t_0) = \mathbf{q}_0 \; and \; \dot{\mathbf{q}}(t_0) = \dot{\mathbf{q}}_0$$

Algorithmic parameters:

U0 Family-Based Algorithmic Parameters:

$$W_1 \Lambda_1 = \frac{1}{1+\rho_\infty^{\mathrm{s}}} \quad , \quad \lambda_1 = 1$$
$$W_2 \Lambda_2 = \frac{1}{2(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_2 = \frac{1}{2}$$
$$W_3 \Lambda_3 = \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_3 = \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})}$$
$$W_1 \Lambda_4 = \frac{1}{1+\rho_\infty^{\mathrm{s}}} \quad , \quad \lambda_4 = 1$$
$$W_2 \Lambda_5 = \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_5 = \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})}$$
$$W_1 \Lambda_6 = \frac{2+\rho_\infty^{\min}+\rho_\infty^{\max}+\rho_\infty^{\mathrm{s}}-\rho_\infty^{\min}\rho_\infty^{\max}\rho_\infty^{\mathrm{s}}}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})}$$

V0 Family-Based Algorithmic Parameters:

$$
\begin{aligned}
W_1\Lambda_1 &= \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \quad , \quad \lambda_1 = 1 \\
W_2\Lambda_2 &= \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \quad , \quad \lambda_2 = \frac{1}{2} \\
W_3\Lambda_3 &= \frac{1}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_3 = \frac{1}{2(1+\rho_\infty^{\mathrm{s}})} \\
W_1\Lambda_4 &= \frac{3+\rho_\infty^{\min}+\rho_\infty^{\max}-\rho_\infty^{\min}\rho_\infty^{\max}}{2(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})} \quad , \quad \lambda_4 = 1 \\
W_2\Lambda_5 &= \frac{2}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})} \quad , \quad \lambda_5 = \frac{1}{1+\rho_\infty^{\mathrm{s}}} \\
W_1\Lambda_6 &= \frac{2+\rho_\infty^{\min}+\rho_\infty^{\max}+\rho_\infty^{\mathrm{s}}-\rho_\infty^{\min}\rho_\infty^{\max}\rho_\infty^{\mathrm{s}}}{(1+\rho_\infty^{\min})(1+\rho_\infty^{\max})(1+\rho_\infty^{\mathrm{s}})}
\end{aligned}
$$

Remark 32 (Algorithm 31)

1. *Algorithm 31 is the single step single solve framework of algorithms encompassing the LMS methods for the second-order system which can be equivalently written in the form of the linear three-step method. The algorithmic parameters of the U0 Family-based and V0 family-based Algorithms are in terms of two principal roots and one spurious root at the high-frequency range which satisfy the following inequality:*

$$0 \leq \rho_\infty^{\mathrm{s}} \leq \rho_\infty^{\min} \leq \rho_\infty^{\max} \leq 1 \tag{9.3}$$

   where $\rho_\infty^{\max}$ is the maximum principal root or spectral radius at the high-frequency range, $\rho_\infty^{\min}$ is the minimum principal root at the high-frequency range, and ρ_∞^{s} is the spurious root at the high-frequency range.

2. ***Implicit GSSSS-2 Framework:*** *When $\eta_1 = \eta_2 = 1$, Algorithm 31 recovers the U0 Family-based and V0 family-based implicit GSSSS Algorithms which are equivalent to Algorithms 17 and 18, respectively. Not only there exist new and optimal algorithms and designs, but also various single step*

single solve algorithms commonly used in the community which primarily belong to the U0 family-based single-field form I-GSSSS-2 algorithmic framework; see Table 4.1. The V0 family-based algorithms are relatively new and novel. As an example, when V0:$(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}})$*=V0:*$(1.0, 1.0, 0.0)$*, i.e.,* $\rho_\infty^{\min} = \rho_\infty^{\max} = 1.0$ *and* $\rho_\infty^{\mathrm{s}} = 0.0$*, the recovered algorithm is called the* ***midpoint rule with the midpoint acceleration (MRP-MPA)****, and it is (spectrally) equivalent to the two-field form symplectic midpoint rule,*

$$\begin{aligned} \mathbf{M}\frac{\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n}{\Delta t} &= \mathbf{f}^{\mathrm{appl}}(\mathbf{q}_{n+1/2}, \mathbf{v}_{n+1/2}, t_{n+1/2}) \\ \frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t} &= \frac{\boldsymbol{\nu}_{n+1} + \boldsymbol{\nu}_n}{2} \end{aligned} \tag{9.4}$$

The basic building blocks of the GSSSS-2 algorithms are shown in Fig. 9.1.

3. ***Explicit GSSSS-2 Framework with the Implicit/Explicit Treatment of the Velocity Term:*** *When* $\eta_2 = 0$ *and* $\eta_1 = 1$*, Algorithm 31 recovers the linear explicit GSSSS algorithmic framework with the* ***implicit treatment of the velocity term****. When* $\eta_2 = 0$ *and* $\eta_1 = 0$*, Algorithm 31 recovers the linear explicit GSSSS algorithmic framework with the* ***explicit treatment of the velocity term****.*

4. ***Time Level Consistency:*** *The algorithmic time level of Algorithm 31 is* $t^* = t_{n+W_1}$ *with*

$$\begin{aligned} \mathbf{q}_n &\approx \mathbf{q}(t_n), \quad \mathbf{q}_{n+1} \approx \mathbf{q}(t_{n+1}) \\ \dot{\mathbf{q}}_n &\approx \dot{\mathbf{q}}(t_n), \quad \dot{\mathbf{q}}_{n+1} \approx \dot{\mathbf{q}}(t_{n+1}) \\ \ddot{\mathbf{q}}_n &\approx \ddot{\mathbf{q}}(t_{n-\phi}), \quad \ddot{\mathbf{q}}_{n+1} \approx \ddot{\mathbf{q}}(t_{n+1-\phi}) \end{aligned} \tag{9.5}$$

where $\phi := W_1(\Lambda_6 - 1)$*; therefore, all* $\mathbf{q}$*,* $\dot{\mathbf{q}}$*, and* $\ddot{\mathbf{q}}$ *are guaranteed to be second-order time accurate.*

First-order Systems

Similarly, employing the ***normalized time weighted residual methodology*** to Eq. (9.2) leads to the generalized single step single solve family of algorithms

which take the representation shown below:

Algorithm 32

Framework of Implicit and Explicit GSSSS Family of Algorithms and Designs for Nonlinear Dynamical Systems in First-order Systems

Integrator:

$$\mathbf{M}\tilde{\mathbf{v}} = \mathbf{f}^{\text{appl}}(\tilde{\mathbf{q}}, t_{n+W_1})$$

where

$$\begin{aligned}\tilde{\mathbf{v}} &= \dot{\mathbf{q}}_n + W_1\Lambda_6\Delta\mathbf{v} \\ \tilde{\mathbf{q}} &= \mathbf{q}_n + W_1\Lambda_4\dot{\mathbf{q}}_n\Delta t + W_2\Lambda_5\Delta\mathbf{v}\Delta t\eta_1\end{aligned}$$

Updates:

$$\begin{aligned}\mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_4\dot{\mathbf{q}}_n\Delta t + \lambda_5\Delta\mathbf{v}\Delta t \\ \dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \Delta\mathbf{v}\end{aligned}$$

Initial condition:

$$\mathbf{q}(t_0) = \mathbf{q}_0$$

Algorithmic parameters:

$$\begin{aligned}W_1\Lambda_4 &= \frac{1}{1+\rho_\infty} \quad , \quad \lambda_4 = 1 \\ W_2\Lambda_5 &= \frac{1}{(1+\rho_\infty)(1+\rho_\infty^{\text{s}})} \quad , \quad \lambda_5 = \frac{1}{1+\rho_\infty^{\text{s}}} \\ W_1\Lambda_6 &= \frac{3+\rho_\infty+\rho_\infty^{\text{s}}-\rho_\infty\rho_\infty^{\text{s}}}{2(1+\rho_\infty)(1+\rho_\infty^{\text{s}})}\end{aligned}$$

Remark 33 (Algorithm 32)

1. *Algorithm 32 is the single step single solve framework of algorithms which can be equivalently written in the form of the linear two-step method. The algorithmic parameters of Algorithms 32 are in terms of two roots at the high-frequency range,*

$$0 \le \rho_\infty^{\text{s}} \le \rho_\infty \le 1 \tag{9.6}$$

where ρ_∞ is the maximum principal root or spectral radius at the high-frequency range and ρ_∞^{s} is the spurious root at the high-frequency range.

2. ***Implicit GSSSS-1 Framework:*** *When $\eta_1 = 1$, Algorithm 32 recovers the implicit GSSSS family of algorithms in the first-order system (**I-GSSSS-1 Algorithms**). When selecting $(\rho_\infty, \rho_\infty^{\mathrm{s}}) = (1, 1)$, we obtain the **Crank-Nicolson method** [62] which is equivalent to the **midpoint/trapezoidal rule**,*

$$\begin{aligned} \mathbf{M}\frac{\dot{\mathbf{q}}_{n+1} + \dot{\mathbf{q}}_n}{2} &= \mathbf{f}^{\mathrm{appl}}(\mathbf{q}_{n+1/2}, t_{n+1/2}) \\ \frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t} &= \frac{\dot{\mathbf{q}}_{n+1} + \dot{\mathbf{q}}_n}{2} \end{aligned} \tag{9.7}$$

*When selecting $(\rho_\infty, \rho_\infty^{\mathrm{s}}) = (0, 0)$, we obtain the **Gear's method** [63],*

$$\begin{aligned} \mathbf{M}\frac{3\dot{\mathbf{q}}_{n+1} - \dot{\mathbf{q}}_n}{2} &= \mathbf{f}^{\mathrm{appl}}(\mathbf{q}_{n+1}, t_{n+1}) \\ \frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t} &= \dot{\mathbf{q}}_{n+1} \end{aligned} \tag{9.8}$$

*When selecting $(\rho_\infty, \rho_\infty^{\mathrm{s}}) = (0, 1)$, we obtain the **MacCormack method** [64]. When selecting $\rho_\infty = \rho_\infty^{\mathrm{s}}$, we obtain **algorithms without selective control feature**; see [65]. When selecting $\rho_\infty \neq \rho_\infty^{\mathrm{s}}$, we obtain **new algorithms and designs with selective control feature**; see [66]. By allowing the two parameters to be equal, i.e., $\rho_\infty = \rho_\infty^{\mathrm{s}}$, the amount of the high frequency damping for the two variables is hence equal (i.e., the high frequency damping is controlled indiscriminately, and not separately) and the framework recovers past development [65] and herein is referred to as the I-GSSSS-1 framework of algorithms without the separable control features. However, the same amount of damping may not be sufficient to suppress the numerical oscillations in the time derivative variable which plays a critical role in several situations. This non-physical instability in the time derivative variable can lead to physically incorrect dynamics of the system for long term simulations. This places a limitation. To overcome this drawback, we allow a more flexible control of the high frequency damping*

by introducing different amounts of numerical dissipation in the two variables which is inherent in the present developments; hence referred herein as the I-GSSSS-1 framework of algorithms with the separable control features. When $(\rho_\infty, \rho_\infty^{\mathrm{s}}) = (1, 0)$*, the resulting algorithm is equivalent to the algorithm in the sense of the single-root system. That is,*

$$\mathbf{M}\frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t} = \mathbf{f}^{\mathrm{appl}}(\mathbf{q}_{n+1/2}, t_{n+1/2}) \tag{9.9}$$

Note that this is the only member which is second-order time accurate and unconditional stable member in the ***single-root system****.*

The basic building blocks of the GSSSS-1 algorithms are shown in Fig. 9.2.

3. ***PCE-GSSSS-1 Framework:*** *When* $\eta_1 = 0$*, Algorithm 32 recovers the predictor-corrector explicit GSSSS family of algorithms in the first-order system (****PCE-GSSSS-1 Algorithms****).*

4. ***Time Level Consistency:*** *The algorithmic time level of Algorithm 32 is* $t^* = t_{n+W_1}$ *with*

$$\begin{aligned} \mathbf{q}_n \approx \mathbf{q}(t_n), \quad \mathbf{q}_{n+1} \approx \mathbf{q}(t_{n+1}) \\ \dot{\mathbf{q}}_n \approx \dot{\mathbf{q}}(t_{n-\phi}), \quad \dot{\mathbf{q}}_{n+1} \approx \dot{\mathbf{q}}(t_{n+1-\phi}) \end{aligned} \tag{9.10}$$

where $\phi := W_1(\Lambda_6 - 1)$*; therefore, both* $\mathbf{q}$ *and* $\dot{\mathbf{q}}$ *are guaranteed to be second-order time accurate.*

9.2 Novel Adaptation Process

Originally, we have individually derived the GSSSS family of algorithms in the second- and first-order systems, i.e., Algorithm 31 and Algorithm 32, by means of the normalized time weighted residual methodologies to the continuous-time equations as shown in Eq. (9.1) and Eq. (9.2), respectively. Here, we show how to readily obtain the I-/PCE-GSSSS family of algorithms in the first-order systems (***I-/PCE-GSSSS-1 algorithms***) directly from the V0 based-family of the

I-/PCE-GSSSS family of algorithms in the second-order systems (***V0 I-/PCE-GSSSS-2 algorithms***) without the need to resort to separately implementing for first-order systems. This is the novelty of the *i*Integration framework. The adaptation process is simply summarized as follows; also see Fig. 9.3:

For variables:
Let (V0 Based-Family of I-/PCE-Algorithm 31 → I-/PCE-Algorithm 32):

$$\begin{aligned} \ddot{\mathbf{q}} &\to \dot{\mathbf{q}} \\ \dot{\mathbf{q}} &\to \mathbf{q} \end{aligned} \tag{9.11}$$

and neglect $\mathbf{q}$ in the V0 I-/PCE-GSSSS-2 algorithms, i.e., treat as a dummy variable.

For algorithmic parameters:
Let (V0 Based-Family of I-/PCE-Algorithm 31 → I-/PCE-Algorithm 32):

$$\begin{aligned} \rho_\infty^{\max} &\to 1 \\ \rho_\infty^{\min} &\to \rho_\infty \\ \rho_\infty^{\mathrm{s}} &\to \rho_\infty^{\mathrm{s}} \end{aligned} \tag{9.12}$$

In linear dynamical systems, the algorithms in the implicit GSSSS family of algorithms have the zero-order overshoot behavior in both configuration and velocity if and only if $\rho_\infty^{\max} = 1$. We notate U0V0 if the algorithms are derived from the U0 family of algorithms. Likewise, we notate V0U0 if the algorithms are derived from the V0 family of algorithms. If $\rho_\infty^{\max} \neq 1$, the U0 family of algorithms have the following features of overshoot behavior, U0V1, i.e., zero-order overshoot and first-order overshoot in the configuration and velocity, respectively; and the V0 family of algorithms have the following features of overshoot behavior, V0U1, i.e., zero-order overshoot and first-order overshoot in the velocity and configuration, respectively. In nonlinear dynamical systems, we employ the same algorithmic parameters. That is, the U0V0-based implicit algorithms are those

algorithms derivable from the U0 family-based Algorithm 31 with $\rho_\infty^{\max} = 1$ and $\eta_1 = \eta_2 = 1$, and the V0U0-based implicit algorithms are those algorithms derivable from the V0 family-based Algorithm 31 with $\rho_\infty^{\max} = 1$ and $\eta_1 = \eta_2 = 1$. If $\rho_\infty^{\max} \neq 1$, the U0V1 family-based and V0U1 family-based algorithms can be obtained from the U0 family-based Algorithm 31 and the V0 family-based Algorithm 31, respectively. Similarly, the U0V0, U0V1, V0U0, and V0U1 family-based predictor-corrector explicit GSSSS algorithms in the second-order systems can be obtained from Algorithm 31 with $\eta_2 = 0$. In linear dynamical systems, note that the GSSSS family of algorithms in the first-order systems have the zero-order overshoot behavior features in both $\mathbf{q}$ and $\dot{\mathbf{q}}$, i.e., U0V0 or V0U0, for any choices of $(\rho_\infty, \rho_\infty^{\mathrm{s}})$.

In Fig. 9.4, we show the relationship between the GSSSS family of algorithms in the second- and first-order systems via the adaptation process.

We show several important adaptation processes directly from the framework of the GSSSS-2 to GSSSS-1 algorithms below. The V0U0(1,1,1) scheme in GSSSS-2 family of algorithms, i.e., MPR-EPA method, leads to U0V0(1,1) scheme in GSSSS-1 family of algorithms, i.e., MPR-EPV method; see Eq. (9.13) and Eq. (9.14). Other typical examples include the following. The V0U0(1,1,0) scheme in GSSSS-2 family of algorithms, i.e., MPR-MPA method, leads to U0V0(1,0) scheme in GSSSS-1 family of algorithms, i.e., MPR-MPV method; see Eq. (9.15) and Eq. (9.16). The V0U0(0,$\rho_\infty^{\max}$,0) scheme in GSSSS-2 family of algorithms leads to U0V0(0,0) scheme in GSSSS-1 family of algorithms, i.e., Gear's method; see Eq. (9.17) and Eq. (9.18). The V0U0(1,1,ρ_∞^{s}) scheme in GSSSS-2 family of algorithms leads to U0V0(1,ρ_∞^{s}) scheme in GSSSS-1 family of algorithms; see Eq. (9.19) and Eq. (9.20). The V0U0($\rho_\infty^{\min}$,1,0) scheme in GSSSS-2 family of algorithms leads to U0V0(ρ_∞,0) scheme in GSSSS-1 family of algorithms; see Eq. (9.21) and Eq. (9.22). The V0U0(ρ_∞,ρ_∞,ρ_∞) scheme in GSSSS-2 family of algorithms, i.e., the counterpart of the three-parameter optimal scheme (note that this scheme is new, and it is not the original three-parameter optimal scheme in [4,5]), leads to U0V0(ρ_∞,ρ_∞) scheme in GSSSS-1 family of algorithms, i.e., the

two-parameter optimal scheme; see Eq. (9.23) and Eq. (9.24).

V0U0: (1,1,1) - MPR-EPA

$$\mathbf{M}\tilde{\mathbf{a}} = \mathbf{f}^{\text{appl}}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}, t_{n+W_1})$$

where

$$\tilde{\mathbf{a}} = \ddot{\mathbf{q}}_n + \frac{1}{2}\Delta\mathbf{a},\ \tilde{\mathbf{v}} = \dot{\mathbf{q}}_n + \frac{\Delta t}{2}\ddot{\mathbf{q}}_n + \frac{\Delta t}{4}\Delta\mathbf{a}$$

$$\tilde{\mathbf{q}} = \mathbf{q}_n + \frac{\Delta t}{2}\dot{\mathbf{q}}_n + \frac{\Delta t^2}{4}\ddot{\mathbf{q}}_n + \frac{\Delta t^2}{8}\Delta\mathbf{a}$$

with

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \Delta t\dot{\mathbf{q}}_n + \frac{\Delta t^2}{2}\ddot{\mathbf{q}}_n + \frac{\Delta t^2}{4}\Delta\mathbf{a} \tag{9.13}$$

$$\dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_n + \Delta t\ddot{\mathbf{q}}_n + \frac{\Delta t}{2}\Delta\mathbf{a}$$

$$\ddot{\mathbf{q}}_{n+1} = \ddot{\mathbf{q}}_{n+1} + \Delta\mathbf{a}$$

and

$$W_1 = \frac{1}{2} \text{ and } \phi := W_1(\Lambda_6 - 1) = 0$$

The algorithm can be cast into the following form:

$$\mathbf{M}\ddot{\mathbf{q}}_{n+1/2} = \mathbf{f}^{\text{appl}}(\mathbf{q}_{n+1/2}, \dot{\mathbf{q}}_{n+1/2}, t_{n+1/2})$$

leads to

U0V0: (1,1) - MPR-EPV

$$\mathbf{M}\tilde{\mathbf{v}} = \mathbf{f}^{\text{appl}}(\tilde{\mathbf{q}}, t_{n+W_1})$$

where

$$\tilde{\mathbf{v}} = \dot{\mathbf{q}}_n + \frac{1}{2}\Delta\mathbf{v},\ \tilde{\mathbf{q}} = \mathbf{q}_n + \frac{\Delta t}{2}\dot{\mathbf{q}}_n + \frac{\Delta t}{4}\Delta\mathbf{v}$$

with

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \Delta t\dot{\mathbf{q}}_n + \frac{\Delta t}{2}\Delta\mathbf{v},\ \dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_{n+1} + \Delta\mathbf{v} \tag{9.14}$$

and

$$W_1 = \frac{1}{2} \text{ and } \phi := W_1(\Lambda_6 - 1) = 0$$

The algorithm can be cast into the following form:

$$\mathbf{M}\ddot{\mathbf{q}}_{n+1/2} = \mathbf{f}^{\text{appl}}(\mathbf{q}_{n+1/2}, t_{n+1/2})$$

V0U0: (1,1,0) - MPR-MPA

$$\mathbf{M}\tilde{\mathbf{a}} = \mathbf{f}^{\text{appl}}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}, t_{n+W_1})$$

where

$$\tilde{\mathbf{a}} = \ddot{\mathbf{q}}_n + \frac{1}{1+\rho_\infty^s}\Delta\mathbf{a},\ \tilde{\mathbf{v}} = \dot{\mathbf{q}}_n + \frac{\Delta t}{2}\ddot{\mathbf{q}}_n + \frac{\Delta t}{2}\Delta\mathbf{a}$$

$$\tilde{\mathbf{q}} = \mathbf{q}_n + \frac{\Delta t}{2}\dot{\mathbf{q}}_n + \frac{\Delta t^2}{4}\ddot{\mathbf{q}}_n + \frac{\Delta t^2}{4}\Delta\mathbf{a}$$

with

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \Delta t\dot{\mathbf{q}}_n + \frac{\Delta t^2}{2}\ddot{\mathbf{q}}_n + \frac{\Delta t^2}{2}\Delta\mathbf{a} \tag{9.15}$$

$$\dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_n + \Delta t\ddot{\mathbf{q}}_n + \Delta t\Delta\mathbf{a}$$

$$\ddot{\mathbf{q}}_{n+1} = \ddot{\mathbf{q}}_{n+1} + \Delta\mathbf{a}$$

and

$$W_1 = \frac{1}{2} \text{ and } \phi := W_1(\Lambda_6 - 1) = \frac{1}{2}$$

The algorithm can be cast into the following form:

$$\mathbf{M}\ddot{\mathbf{q}}_{n+1} = \mathbf{f}^{\text{appl}}(\mathbf{q}_{n+1}, \dot{\mathbf{q}}_{n+1}, t_{n+1})$$

leads to

U0V0: (1,0) - MPR-MPV

$$\mathbf{M}\tilde{\mathbf{v}} = \mathbf{f}^{\text{appl}}(\tilde{\mathbf{q}}, t_{n+W_1})$$

where

$$\tilde{\mathbf{v}} = \dot{\mathbf{q}}_n + \Delta\mathbf{v},\ \tilde{\mathbf{q}} = \mathbf{q}_n + \frac{\Delta t}{2}\dot{\mathbf{q}}_n + \frac{\Delta t}{2}\Delta\mathbf{v}$$

with

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \Delta t\dot{\mathbf{q}}_n + \Delta t\Delta\mathbf{v},\ \dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_{n+1} + \Delta\mathbf{v} \tag{9.16}$$

and

$$W_1 = \frac{1}{2} \text{ and } \phi := W_1(\Lambda_6 - 1) = \frac{1}{2}$$

The algorithm can be cast into the following form:

$$\mathbf{M}\dot{\mathbf{q}}_{n+1} = \mathbf{f}^{\text{appl}}(\mathbf{q}_{n+1}, t_{n+1})$$

V0U0: (0,$\rho_\infty^{\max}$,0)

$$\mathbf{M}\tilde{\mathbf{a}} = \mathbf{f}^{\text{appl}}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}, t_{n+1})$$

where

$$\tilde{\mathbf{a}} = \ddot{\mathbf{q}}_n + \frac{2+\rho_\infty^{\max}}{1+\rho_\infty^{\max}}\Delta\mathbf{a},\ \tilde{\mathbf{v}} = \dot{\mathbf{q}}_n + \Delta t\ddot{\mathbf{q}}_n + \frac{3+\rho_\infty^{\max}}{2(1+\rho_\infty^{\max})}\Delta t\Delta\mathbf{a}$$

$$\tilde{\mathbf{q}} = \mathbf{q}_n + \Delta t\dot{\mathbf{q}}_n + \frac{\Delta t^2}{2}\ddot{\mathbf{q}}_n + \frac{\Delta t^2}{1+\rho_\infty^{\max}}\Delta\mathbf{a}$$

with

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \Delta t\dot{\mathbf{q}}_n + \frac{\Delta t^2}{2}\ddot{\mathbf{q}}_n + \frac{\Delta t^2}{2(1+\rho_\infty^{\max})}\Delta\mathbf{a} \tag{9.17}$$

$$\dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_n + \Delta t\ddot{\mathbf{q}}_n + \frac{3+\rho_\infty^{\max}}{2(1+\rho_\infty^{\max})}\Delta t\Delta\mathbf{a}$$

$$\ddot{\mathbf{q}}_{n+1} = \ddot{\mathbf{q}}_{n+1} + \Delta\mathbf{a}$$

and

$$W_1 = 1 \text{ and } \phi := W_1(\Lambda_6 - 1) = \frac{1}{1+\rho_\infty^{\max}} \in [\frac{1}{2}, 1]$$

The algorithm can be cast into the following form:

$$\mathbf{M}\left[\ddot{\mathbf{q}}_n + \frac{2+\rho_\infty^{\max}}{1+\rho_\infty^{\max}}\Delta\mathbf{a}\right] = \mathbf{f}^{\text{appl}}(\mathbf{q}_{n+1}, \dot{\mathbf{q}}_{n+1}, t_{n+1})$$

leads to

U0V0: (0,0) - Gear's Method

$$\mathbf{M}\tilde{\mathbf{v}} = \mathbf{f}^{\text{appl}}(\tilde{\mathbf{q}}, t_{n+1})$$

where

$$\tilde{\mathbf{v}} = \dot{\mathbf{q}}_n + \frac{3}{2}\Delta\mathbf{v},\ \tilde{\mathbf{q}} = \mathbf{q}_n + \Delta t\dot{\mathbf{q}}_n + \Delta t\Delta\mathbf{v}$$

with

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \Delta t\dot{\mathbf{q}}_n + \Delta t\Delta\mathbf{v},\ \dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_{n+1} + \Delta\mathbf{v} \tag{9.18}$$

and

$$W_1 = 1 \text{ and } \phi := W_1(\Lambda_6 - 1) = \frac{1}{2}$$

The algorithm can be cast into the following form:

$$\mathbf{M}\left[\dot{\mathbf{q}}_n + \frac{3}{2}\Delta\mathbf{v}\right] = \mathbf{f}^{\text{appl}}(\mathbf{q}_{n+1}, t_{n+1})$$

V0U0: (1,1,ρ^{s}_∞)

$$\mathbf{M}\tilde{\mathbf{a}} = \mathbf{f}^{\mathrm{appl}}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}, t_{n+W_1})$$

where

$$\tilde{\mathbf{a}} = \ddot{\mathbf{q}}_n + \frac{1}{1+\rho^{\mathrm{s}}_\infty}\Delta\mathbf{a},\ \tilde{\mathbf{v}} = \dot{\mathbf{q}}_n + \frac{\Delta t}{2}\ddot{\mathbf{q}}_n + \frac{\Delta t}{2(1+\rho^{\mathrm{s}}_\infty)}\Delta\mathbf{a}$$

$$\tilde{\mathbf{q}} = \mathbf{q}_n + \frac{\Delta t}{2}\dot{\mathbf{q}}_n + \frac{\Delta t^2}{4}\ddot{\mathbf{q}}_n + \frac{\Delta t^2}{4(1+\rho^{\mathrm{s}}_\infty)}\Delta\mathbf{a}$$

with

$$\begin{aligned}
\mathbf{q}_{n+1} &= \mathbf{q}_n + \Delta t\dot{\mathbf{q}}_n + \frac{\Delta t^2}{2}\ddot{\mathbf{q}}_n + \frac{\Delta t^2}{2(1+\rho^{\mathrm{s}}_\infty)}\Delta\mathbf{a} \\
\dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \Delta t\ddot{\mathbf{q}}_n + \frac{\Delta t}{1+\rho^{\mathrm{s}}_\infty}\Delta\mathbf{a} \\
\ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_{n+1} + \Delta\mathbf{a}
\end{aligned} \tag{9.19}$$

and

$$W_1 = \frac{1}{2} \text{ and } \phi := W_1(\Lambda_6 - 1) = \frac{1-\rho^{\mathrm{s}}_\infty}{2(1+\rho^{\mathrm{s}}_\infty)} \in [0, \frac{1}{2}]$$

The algorithm can be cast into the following form:

$$\mathbf{M}\left[\ddot{\mathbf{q}}_n + \frac{1}{1+\rho^{\mathrm{s}}_\infty}\Delta\mathbf{a}\right] = \mathbf{f}^{\mathrm{appl}}(\mathbf{q}_{n+1/2}, \dot{\mathbf{q}}_{n+1/2}, t_{n+1/2})$$

leads to

U0V0: (1,ρ^{s}_∞)

$$\mathbf{M}\tilde{\mathbf{v}} = \mathbf{f}^{\mathrm{appl}}(\tilde{\mathbf{q}}, t_{n+W_1})$$

where

$$\tilde{\mathbf{v}} = \dot{\mathbf{q}}_n + \frac{1}{1+\rho^{\mathrm{s}}_\infty}\Delta\mathbf{v},\ \tilde{\mathbf{q}} = \mathbf{q}_n + \frac{\Delta t}{2}\dot{\mathbf{q}}_n + \frac{\Delta t}{2(1+\rho^{\mathrm{s}}_\infty)}\Delta\mathbf{v}$$

with

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \Delta t\dot{\mathbf{q}}_n + \frac{\Delta t}{1+\rho^{\mathrm{s}}_\infty}\Delta\mathbf{v},\ \dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_{n+1} + \Delta\mathbf{v} \tag{9.20}$$

and

$$W_1 = \frac{1}{2} \text{ and } \phi := W_1(\Lambda_6 - 1) = \frac{1-\rho^{\mathrm{s}}_\infty}{2(1+\rho^{\mathrm{s}}_\infty)} \in [0, \frac{1}{2}]$$

The algorithm can be cast into the following form:

$$\mathbf{M}\left[\dot{\mathbf{q}}_n + \frac{1}{1+\rho^{\mathrm{s}}_\infty}\Delta\mathbf{v}\right] = \mathbf{f}^{\mathrm{appl}}(\mathbf{q}_{n+1/2}, t_{n+1/2})$$

V0U0: ($\rho_\infty^{\min}$,1,0)

$$\mathbf{M}\tilde{\mathbf{a}} = \mathbf{f}^{\text{appl}}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}, t_{n+W_1})$$

where

$$\tilde{\mathbf{a}} = \ddot{\mathbf{q}}_n + \frac{3+\rho_\infty^{\min}}{2(1+\rho_\infty^{\min})}\Delta\mathbf{a},\ \tilde{\mathbf{v}} = \dot{\mathbf{q}}_n + \frac{\Delta t}{1+\rho_\infty^{\min}}\ddot{\mathbf{q}}_n + \frac{\Delta t}{1+\rho_\infty^{\min}}\Delta\mathbf{a}$$

$$\tilde{\mathbf{q}} = \mathbf{q}_n + \frac{\Delta t}{1+\rho_\infty^{\min}}\dot{\mathbf{q}}_n + \frac{\Delta t^2}{2(1+\rho_\infty^{\min})}\ddot{\mathbf{q}}_n + \frac{\Delta t^2}{2(1+\rho_\infty^{\min})}\Delta\mathbf{a}$$

with

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \Delta t\dot{\mathbf{q}}_n + \frac{\Delta t^2}{2}\ddot{\mathbf{q}}_n + \frac{\Delta t^2}{2}\Delta\mathbf{a} \tag{9.21}$$

$$\dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_n + \Delta t\ddot{\mathbf{q}}_n + \Delta t\Delta\mathbf{a}$$

$$\ddot{\mathbf{q}}_{n+1} = \ddot{\mathbf{q}}_{n+1} + \Delta\mathbf{a}$$

and

$$W_1 = \frac{1}{1+\rho_\infty^{\min}} \text{ and } \phi := W_1(\Lambda_6 - 1) = \frac{1}{2}$$

The algorithm can be cast into the following form:

$$\mathbf{M}\left[\ddot{\mathbf{q}}_n + \frac{3+\rho_\infty^{\min}}{2(1+\rho_\infty^{\min})}\Delta\mathbf{a}\right] = \mathbf{f}^{\text{appl}}(\mathbf{q}_{n+W_1}, \dot{\mathbf{q}}_{n+W_1}, t_{n+W_1})$$

leads to

U0V0: (ρ_∞,0)

$$\mathbf{M}\tilde{\mathbf{v}} = \mathbf{f}^{\text{appl}}(\tilde{\mathbf{q}}, t_{n+W_1})$$

where

$$\tilde{\mathbf{v}} = \dot{\mathbf{q}}_n + \frac{3+\rho_\infty}{2(1+\rho_\infty)}\Delta\mathbf{v},\ \tilde{\mathbf{q}} = \mathbf{q}_n + \frac{\Delta t}{1+\rho_\infty}\dot{\mathbf{q}}_n + \frac{\Delta t}{1+\rho_\infty}\Delta\mathbf{v}$$

with

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \Delta t\dot{\mathbf{q}}_n + \Delta t\Delta\mathbf{v},\ \dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_{n+1} + \Delta\mathbf{v} \tag{9.22}$$

and

$$W_1 = \frac{1}{1+\rho_\infty} \text{ and } \phi := W_1(\Lambda_6 - 1) = \frac{1}{2}$$

The algorithm can be cast into the following form:

$$\mathbf{M}\left[\dot{\mathbf{q}}_n + \frac{3+\rho_\infty}{2(1+\rho_\infty)}\Delta\mathbf{v}\right] = \mathbf{f}^{\text{appl}}(\mathbf{q}_{n+1/2}, t_{n+1/2})$$

V0U0: (ρ_∞,ρ_∞,ρ_∞) where $\rho_\infty := \rho_\infty^{\min} = \rho_\infty^{\max} = \rho_\infty^{\mathrm{s}} \in [0,1]$

$$\mathbf{M}\tilde{\mathbf{a}} = \mathbf{f}^{\mathrm{appl}}(\tilde{\mathbf{q}}, \tilde{\mathbf{v}}, t_{n+W_1})$$

where

$$\tilde{\mathbf{a}} = \ddot{\mathbf{q}}_n + \frac{2-\rho_\infty}{1+\rho_\infty}\Delta\mathbf{a},\ \tilde{\mathbf{v}} = \dot{\mathbf{q}}_n + \frac{3-\rho_\infty}{2(1+\rho_\infty)}\Delta t\ddot{\mathbf{q}}_n + \frac{2}{(1+\rho_\infty)^3}\Delta t\Delta\mathbf{a}$$

$$\tilde{\mathbf{q}} = \mathbf{q}_n + \frac{3-\rho_\infty}{2(1+\rho_\infty)}\Delta t\dot{\mathbf{q}}_n + \frac{\Delta t^2}{(1+\rho_\infty)^2}\ddot{\mathbf{q}}_n + \frac{\Delta t^2}{(1+\rho_\infty)^3}\Delta\mathbf{a}$$

with

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \Delta t\dot{\mathbf{q}}_n + \frac{\Delta t^2}{2}\ddot{\mathbf{q}}_n + \frac{\Delta t^2}{2(1+\rho_\infty)}\Delta\mathbf{a}$$

$$\dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_n + \Delta t\ddot{\mathbf{q}}_n + \frac{\Delta t}{1+\rho_\infty}\Delta\mathbf{a}$$

$$\ddot{\mathbf{q}}_{n+1} = \ddot{\mathbf{q}}_{n+1} + \Delta\mathbf{a}$$

(9.23)

and

$$W_1 = \frac{3-\rho_\infty}{2(1+\rho_\infty)} \in [\frac{1}{2}, \frac{3}{2}] \text{ and } \phi := W_1(\Lambda_6 - 1) = \frac{1-\rho_\infty}{2(1+\rho_\infty)} \in [0, \frac{1}{2}]$$

The algorithm can be cast into the following form:

$$\mathbf{M}\left[\ddot{\mathbf{q}}_n + \frac{3-\rho_\infty}{2(1+\rho_\infty)}\Delta\mathbf{a}\right] = \mathbf{f}^{\mathrm{appl}}(\mathbf{q}_{n+W_1}, \dot{\mathbf{q}}_{n+W_1}, t_{n+W_1})$$

where

$$\tilde{\mathbf{q}} = \mathbf{q}_n + \frac{2}{(1+\rho_\infty)^2}\Delta\mathbf{q} - \frac{(1-\rho_\infty)^2}{2(1+\rho_\infty)^2}\Delta t\dot{\mathbf{q}}_n$$

$$\tilde{\mathbf{v}} = \dot{\mathbf{q}}_n + \frac{2}{(1+\rho_\infty)^2}\Delta\mathbf{v} - \frac{(1-\rho_\infty)^2}{2(1+\rho_\infty)^2}\Delta t\ddot{\mathbf{q}}_n$$

leads to

U0V0: (ρ_∞,ρ_∞) where $\rho_\infty = \rho_\infty^{\mathrm{s}} \in [0,1]$

$$\mathbf{M}\tilde{\mathbf{v}} = \mathbf{f}^{\mathrm{appl}}(\tilde{\mathbf{q}}, t_{n+W_1})$$

where

$$\tilde{\mathbf{v}} = \dot{\mathbf{q}}_n + \frac{3-\rho_\infty}{2(1+\rho_\infty)}\Delta\mathbf{v},\ \tilde{\mathbf{q}} = \mathbf{q}_n + \frac{\Delta t}{1+\rho_\infty}\dot{\mathbf{q}}_n + \frac{\Delta t}{(1+\rho_\infty)^2}\Delta\mathbf{v}$$

with

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \Delta t\dot{\mathbf{q}}_n + \frac{\Delta t}{1+\rho_\infty}\Delta\mathbf{v},\ \dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_{n+1} + \Delta\mathbf{v} \tag{9.24}$$

and

$$W_1 = \frac{1}{1+\rho_\infty} \in [\frac{1}{2}, 1] \text{ and } \phi := W_1(\Lambda_6 - 1) = \frac{1-\rho_\infty}{2(1+\rho_\infty)} \in [0, \frac{1}{2}]$$

The algorithm can be cast into the following form:

$$\mathbf{M}\left[\dot{\mathbf{q}}_n + \frac{3-\rho_\infty}{2(1+\rho_\infty)}\Delta\mathbf{v}\right] = \mathbf{f}^{\mathrm{appl}}(\mathbf{q}_{n+W_1}, t_{n+W_1})$$

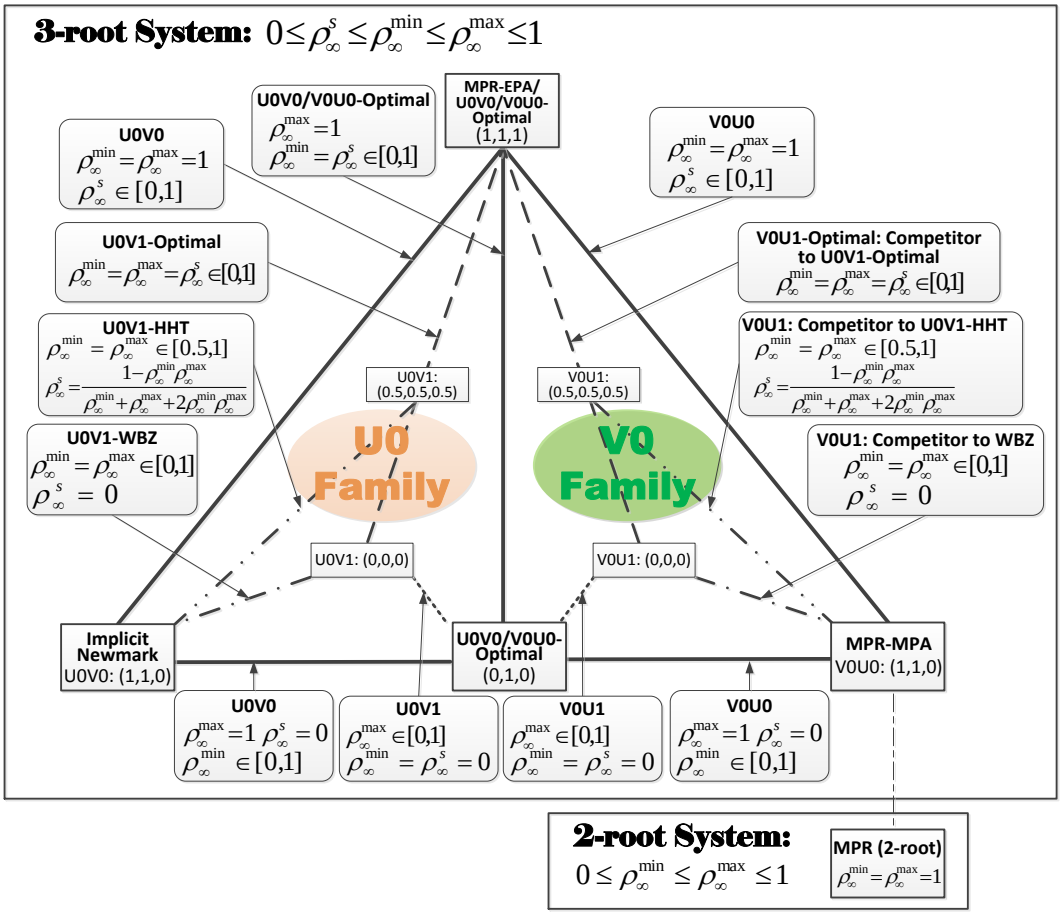


Figure 9.1: Basic Building Blocks of I-GSSSS Family of algorithms in second-order systems

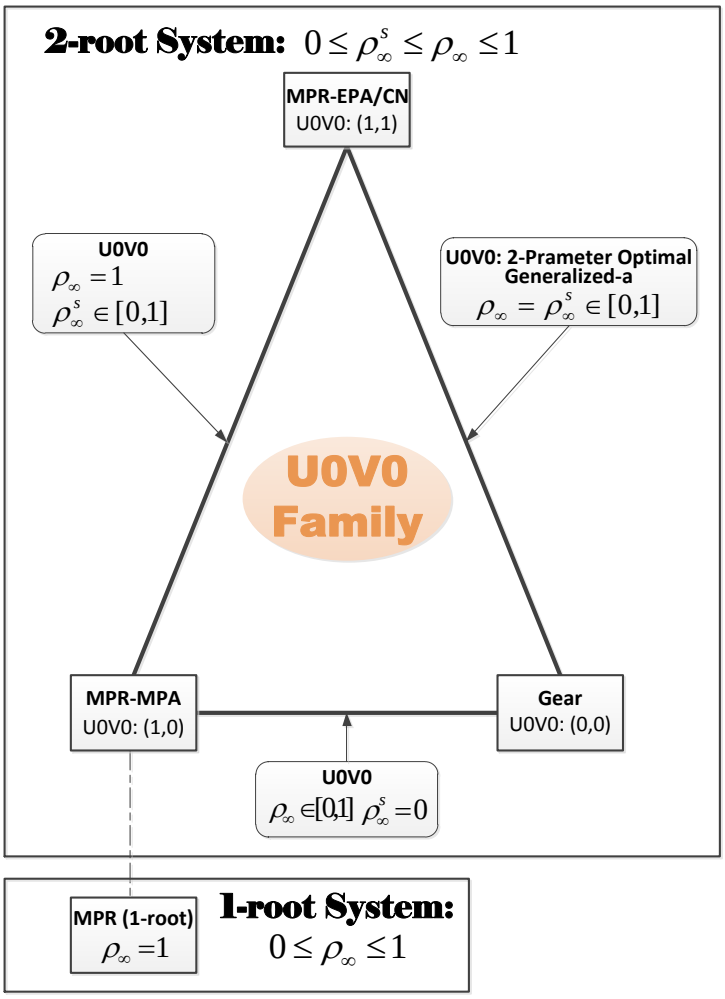


Figure 9.2: Basic Building Blocks of I-GSSSS Family of algorithms in first-order systems

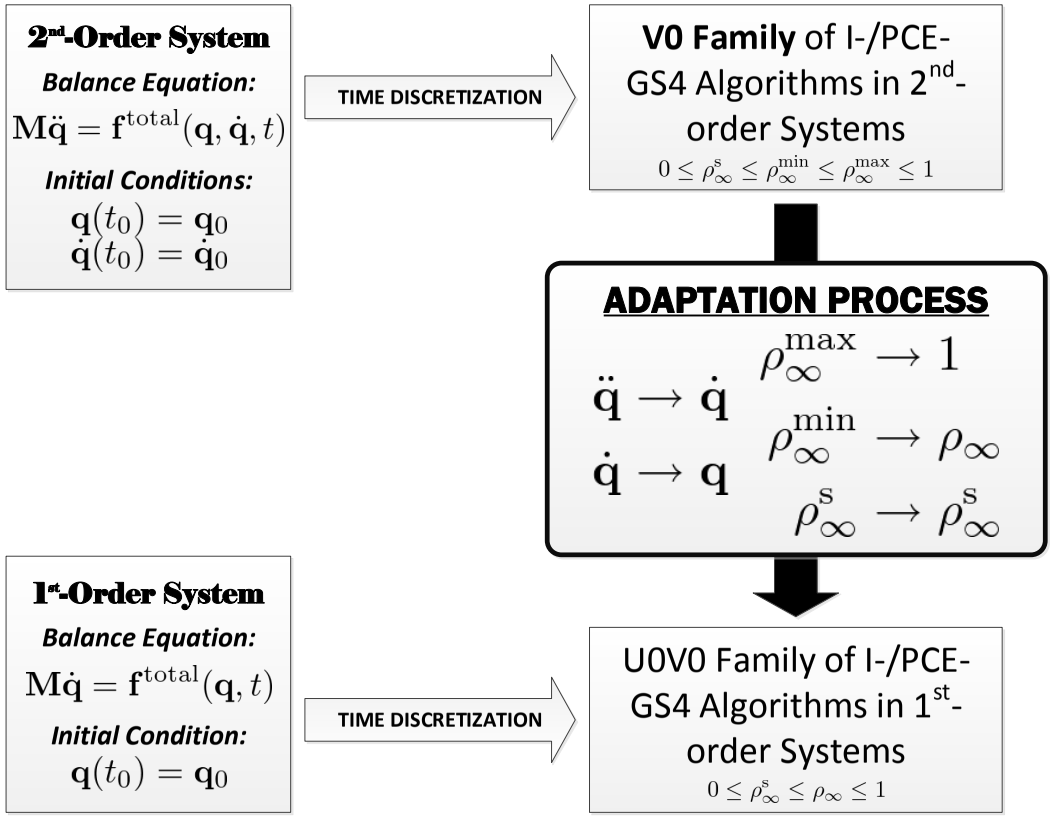


Figure 9.3: iINTEGRATION framework: Adaptation Process

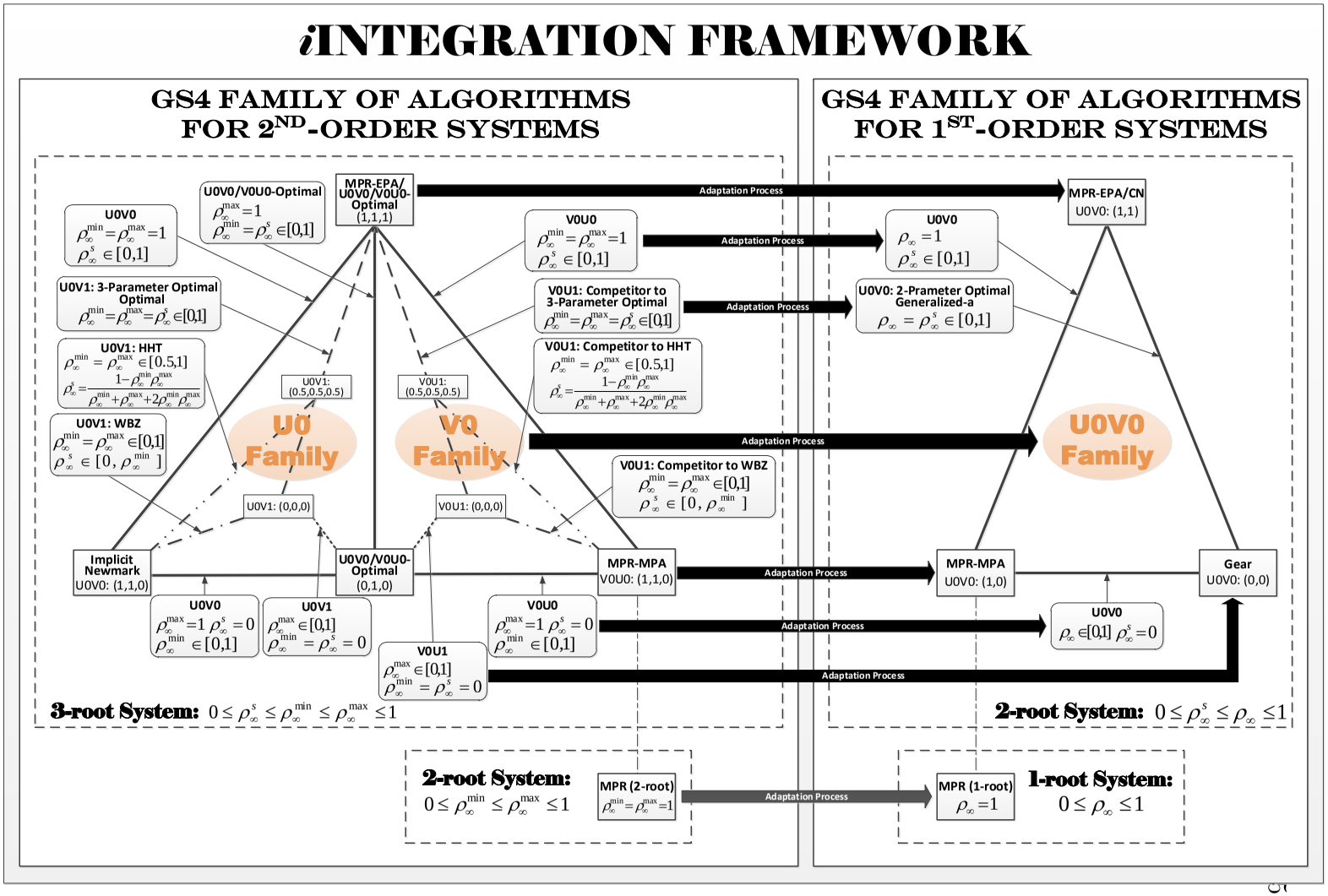


Figure 9.4: *i*INTEGRATION framework: Overview

Chapter 10

Conclusions

The design of an *i*Integration framework and architecture encompassing a wide class of single step and single solve algorithms mostly dealing with those termed as Linear Multi Step (LMS) methods that is applicable to both second order and first order systems as well was presented. The unified framework basically encompasses: (i) Implicit and explicit GSSSS family of algorithms in the single- and two-field forms in the second-order time system, and (ii) Implicit and explicit GSSSS family of algorithms in the first-order time system. Additionally, a general framework of the explicit time integration schemes in the second-order time system has also been proposed.

The basic idea of designing the time integration schemes was based upon and emanates from the time weighted residual methodology. The major developments included: (i) All the designs of the resulting developments are strictly second-order time accurate which is an important design concern (first order time accurate methods are not looked upon favorably); (ii) all algorithms and designs possess a consistent time level in the time discretized equations which is not well understood to-date; (iii) Linear dynamics and algorithms and designs were first addressed; the details of the stability, accuracy and overshoot were presented; (iv) the design of implicit frameworks and the corresponding predictor-corrector

explicit algorithms and designs were then described; (v) extensions of linear dynamics algorithms to nonlinear dynamics applications was addressed next using a novel normalized time weighted residual methodology and leading to those termed as symplectic-momentum conserving and energy momentum conserving designs; and (iv) lastly, a new and novel *i*Integration framework that is applicable to both second order systems and first order systems was finally designed for applicability to general computational engineering and science problems. The novel and significant advantage of this framework and architecture are that the analyst needs to simply use a single unified framework, and consequently has at hand a wide variety of algorithms and choices that are readily available including optimal algorithm designs.

All schemes presented in this thesis possess the much desired second-order time accuracy. The implicit schemes that are unconditionally stable and of zero-th order overshoot in the configuration and/or velocity in the sense of linear dynamical systems were mostly focused upon (higher order overshoot algorithms have not been discussed as they are not competitive. The relations with traditional developments as related to those termed as variational algorithms and exact energy-momentum conserving algorithms within the present time integration framework have also have been carefully discussed. Finally a wide variety of N-body and continuum elastodynamics problems were presented which confirm the theoretical developments.

References

[1] J. E. Marsden and M. West. Discrete Mechanics and Variational Integrators. *Acta Numerica*, 10:357–514, 2001.

[2] X. Zhou, D. Sha, and K. K. Tamma. An Arbitrary Reference Configuration (ARC) Elasticity Theory and the Corresponding Stress Update Formulation for Finite Deformation Analysis. In *46st AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference and Exhibit*, number AIAA-2005-1921, Austin, TX, April 18-21 2005. AIAA.

[3] J. Chung and G. Hulbert. A Time Integration Method for Structural Dynamics With Improved Numerical Dissipation: The Generalized α-Method. *Journal of Applied Mechanics*, 30:371–375, 1993.

[4] H. P. Shao. The Studying on the Direct Time Integration Algorithms for Structural Dynamics Response. Master's thesis, Zheng Jiang University, 1987.

[5] H. P. Shao and C. W. Cai. The Direct Integration Three-Parameters Optimal Schemes for Structural Dynamics (in English). In *Proceeding of the International Conference: Machine Dynamics and Engineering Applications*, pages C16–C20. Xi'an Jiaotong University Press (see also Chinese Journal of Applied Mechanics (in Chinese), Volume 5, No.4, Dec. 1988), 1988.

[6] D. T. Greenwood. *Classical Dynamics*. Prentice-Hall, Reading, Mass., 1977.

[7] J. Har and K. K. Tamma. *Advances in Computational Dynamics of Particles, Materials and Structures*. John Wiley, Chichester,UK, 2012.

[8] K. K. Tamma, J. Har, X. Zhou, M. Shimada, and A. Hoitink. An Overview and Recent Advances in Vector and Scalar Formalisms: Space/Time Discretizations in Computational Dynamics: A Unified Approach. *Arch Comput Methods Eng*, 2011 (DOI 10.1007/s11831-011-9060-y).

[9] L.A. Pars. *A Treatise on Analytical Dynamics*. John Wiley & Sons, New York, 1965.

[10] C. Truesdell and R. A. Toupin. *The Classical Field Theories*. Handbuch der Physik, III/1. Springer-Verlag, Berlin, 1960.

[11] C. Truesdell. *A First Course in Rational Continuum Mechanics* . Academic Press, Boston, second edition edition, 1991.

[12] Lawrence E. Malvern. *Introduction to the Mechanics of a Continuous Medium*. Prentice-Hall, New Jersey, 1969.

[13] J. E. Marsden and T. J. R. Hughes. *Mathematical Foundations of Elasticity*. Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1983.

[14] G. A. Holzapfel. *Nonlinear Solid Mechanics: A Continuum Approach for Engineering*. John Wiley & Sons, New York, 2000.

[15] T. Belytschko, W. K. Liu, and B. Moran. *Nonlinear Finite Elements for Continua and Structures*. John Wiley & Sons, 2000.

[16] I. G. Bubnov. Report on the Works of Prof. Timoshenko Which were Awarded the Zhuranskii Prize. *Symposium of the Institute of Communication Engineers, No. 81, All Union Special Planning Office*, 1913.

[17] B. G. Galerkin. Rods and Plates. Series in Some Problems of Elastic Equilibrium of Rods and Plates. *Vestn. Inzh, Tech*, 19, 1915.

[18] Stephen H. Crandall. *Engineering Analysis*. McGraw-Hill, New York, 1956.

[19] S. G. Mikhlin. *Variational Methods in Mathematical Physics*. Macmillan, New York, 1964.

[20] B. A. Finlayson. *The Method of Weighted Residuals and Variational Principles*. Academic Press, 1972.

[21] T. J. R. Hughes. *The Finite Element Method, Linear Static and Dynamic Finite Element Analysis*. Prentice-Hall, Englewood Cliffs, New Jersey, 1987.

[22] J. N. Reddy. *An Introduction to the Finite Element Method*. McGraw-Hill, New York, 2006.

[23] C. Lanczos. *The Variational Principles of Mechanics*. University of Toronto Press, Toronto, CA, 1970.

[24] H. Goldstein. *Classical Mechanics*. Addison Wesley, San Francisco, 2002.

[25] Mark D. Ardema. *Analytical Dynamics : Theory and Applications*. Kluwer Academic/Plenum Publishers, New York, 2005.

[26] K. Rektorys. *Variational Methods in Mathematics, Science and Engineering*. D. Reidel, Dordrecht, Holland, Boston, 2 edition, 1980.

[27] J. N. Reddy. *Applied Functional Analysis and Variational Methods in Engineering* . McGraw-Hill, New York, 1986.

[28] K. K. Tamma, X. Zhou, and D. Sha. The Time Dimension: A Theory Towards the Evolution, Classification, Characterization and Design of Computational Algorithms for Transient/ Dynamic Applications. *Archives of Computational Methods in Engineering*, 7(2):67–290, 2000.

[29] J. D. Lambert. *Computational Methods in Ordinary Differential Equations*. John Wiley & Sons, New York, 1977.

[30] E. J. Routh. *A Treatise on the Stability of a Given State of Motion*. Macmillan, London, 1877.

[31] E. J. Routh. *The Advanced Part of a Treaties on the Dynamics of a Rigid Body*. Macmillan, 6th edition, 1905. reprinted (1959) Dover, New York.

[32] A. Hurwitz. Über die Bedingungen, Unter Welchen Eine Gleichung Nur Wurzeln Mit Negativen Reellen Teilen Besitzt. *Mathematische Annalen*, 46:273–284, 1895.

[33] N. M. Newmark. A Method of Computation for Structural Dynamics. *Journal for American Society of Civil Engineers*, 1:67–94, 1959.

[34] X. Zhou and K. K. Tamma. Design, Analysis, and Synthesis of Generalized Single Step Single Solve and Optimal Algorithms for Structural Dynamics. *International Journal for Numerical Methods in Engineering*, 59:597–668, 2004.

[35] W. L. Wood, M. Bossak, and O. C. Zienkiewicz. An Alpha Modification of Newmark's Method. *International Journal for Numerical Methods in Engineering*, 15:1562–1566, 1980.

[36] E. L. Wilson. *A Computer Program for Dynamic Stress Analysis of Underground Structures*. SESM, University of California, Berkeley, 1968.

[37] G. Dahlquist. A Special Stability Problem for Linear Multistep Methods. *BIT*, 3:27–43, 1963.

[38] K. K. Tamma and R. R. Namburu. A New Finite Element Based Lax-Wendroff/Taylor-Galerkin Methodology for Computational Dynamics. *Computer Methods in Applied Mechanics and Engineering*, 71:137–150, 1988.

[39] K. K. Tamma and R. R. Namburu. Applicability and Evaluation of An Implicit Self-Starting Unconditionally Stable Methodology for Dynamics of Structures. *Computers & Structures*, 34:835–842, 1990.

[40] K. K. Tamma and Raju R. Namburu. Computational Approaches With Applications to Non-classical and Classical Thermomechanical Problems. *Applied Mechanics Reviews*, 50:514, 1997.

[41] W. C. Swope, H. C. Anderson, P. H. Berens, and K. R. Wilson. A computer simulation method for the calculation of equilibrium constants for the formation of physical clusters of molecules: Application to small water clusters. *The Journal of Chemical Physics*, 76:648, 1882.

[42] G. M. Hulbert and J. Chung. Explicit Time Integration Algorithms for Structural Dynamics with Optimal Numerical Dissipation. *Comput. Methods Appl. Mech. Engrg*, 137:175, 1996.

[43] X. Zhou S. Masuri, A. Hoitink and K.K. Tamma. Algorithms by Design: Part II - A Novel Normalized Time Weighted Residual Methodology and Design of a Family of Symplectic-Momentum Conserving Algorithms for Nonlinear Structural Dynamics. *International Journal for Computational Methods in Engineering Science and Mechanics*, 1(1), 2009.

[44] X. Zhou S. Masuri, A. Hoitink and K.K. Tamma. Algorithms by Design: Part III A Novel Normalized Time Weighted Residual Methodology and Design of Optimal Symplectic-Momentum Based Controllable Numerical Dissipative Algorithms for Nonlinear Structural Dynamics. *International Journal for Computational Methods in Engineering Science and Mechanics*, 10(1550-2295), 2009.

[45] S. Masuri, A. Hoitink, X. Zhou, and K. K. Tamma. Algorithms by Design: A New Normalized Time-Weighted Residual Methodology and Design Leading to a Family of EnergyMomentum Conserving Algorithms for Non-linear Structural Dynamics. *International Journal of Numerical Methods in Engineering*, 79:1094–1146, 2009.

[46] C. Kane, J. E. Marsden, M. Ortiz, and M. West. Variational Integrators and the Newmark Algorithm for Conservative and Dissipative Mechanical Systems. *International Journal for Numerical Methods in Engineering*, 49:1295–1325, 2000.

[47] K. J. Bathe and M. M. I. Baig. On a Composite Implicit Time Integration Procedure for Nonlinear Dynamics. *Computer and Structures*, 83:2513–2534, 2005.

[48] K. J. Bathe. Conserving Energy and Momentum in Nonlinear Dynamics: A Simple Implicit Time Integration Scheme. *Computer and Structures*, 85:437–445, 2007.

[49] D. Greenspan. *Discrete Models*. Addison-Wesley, Reading, Mass., 1973.

[50] R. A. Labudde and D. Greenspan. Energy and Momentum Conserving Methods of Arbitrary Order for the Numerical Integration of Equations of Motion Part I. *Numerisch Mathematik*, 25:323–346, 1976.

[51] R. A. Labudde and D. Greenspan. Energy and Momentum Conserving Methods of Arbitrary Order for the Numerical Integration of Equations of Motion Part II. *Numerisch Mathematik*, 26:1–16, 1976.

[52] J. C. Simo, N. Tarnow, and K. K. Wong. Exact Energy-Momentum Conserving Algorithms and Symplectic Schemes for Nonlinear Dynamics. *Computer Methods in Applied Mechanics and Engineering*, 100(1):63–116, 1992.

[53] P. Betsch and P. Steinmann. Conservation properties of a time FE method. Part I: time-stepping Schemes for N-body problems. *International Journal for Numerical Methods in Engineering*, 49:599–638, 2000.

[54] E. Graham, G. Jelenic, and M. A. Crisfield. A Note on the Equivalence of Two Recent Time-Integration Schemes for N-Boday Problems. *Communications in Numerical Methods in Engineering*, 18:615–620, 2002.

[55] J.C. Simo and N. Tarnow. The Discrete Energy-Momentum Method. Conserving Algorithms for Nonlinear Elastodynamics. *Journal of Applied Mathematics and Physics*, 43:757–792, 1992.

[56] J. M. Sanz-Serna and M. P. Calvo. *Numerical Hamiltonian Problems*. Chapman & Hall, London SE1 8HN, UK, 1994.

[57] E. Hairer, C. Lubich, and G. Wanner. *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*. Springer-Verlag, New York, 2002.

[58] A. Hoitink, S. Masuri, X. Zhou, and K. K. Tamma. Algorithms by Design: Part I - On the Hidden Point Collocation Within LMS Methods and Implications for Nonlinear Dynamics Applications. *International Journal for Computational Methods in Engineering Science and Mechanics*, 8:383–407, 2008.

[59] P. Betsch and P. Steinmann. Conservation Properties of a Time FE method - Part II: Time-Stepping Schemes for Non-linear Elastodynamics. *International Journal for Numerical Methods in Engineering*, 50:1931–1955, 2001.

[60] J. C. Simo and O. Gonzalez. Recent Results on the Numerical Integration of Infinite-Dimensional Hamiltonian Systems. *IN: Recent Developments in Finite Element Analysis, CIMNE*, pages 255–271, 1994.

[61] O. Gonzalez. Exact Energy and Momentum Conserving Algorithms for General Models in Nonlinear Elasticity. *Computer Methods in Applied Mechanics and Engineering*, 190:1763–1783, 2000.

[62] J. Crank and P. Nicolson. A Practical Method for the Numerical Evaluation of Solutions of Partial Differential Equations of the Heat Conduction Type. *Proc. Cambridge Phil. Soc.*, 43:50, 1947.

[63] C. W. Gear, editor. *Numerical Initial Value Problems in Ordinary Differential Equations*. Prentice-Hall, Englewood Cliffs, NJ, 1971.

[64] R. W. MacCormack. The Effect of Viscosity in Hypervelocity Impact Cratering. *AIAA*, 69:354, 1969.

[65] Kenneth E. Jansen, Christian H. Whiting, and Gregory M. Hulbert. A Generalized-α Method for Integrating the Filtered Naiver-Stokes Equations with a Stabilized Finite Element Method. *Comput. Methods Appl. Mech. Engrg*, 190:305–319, 2000.

[66] X. Zhou S. Masuri, M. Sellier and K. K. Tamma. Design of order-preserving algorithms for transient first-order systems with controllable numerical dissipation. *International Journal for Numerical Methods in Engineering*, 20011.

[67] T. A. Laursen and X. N. Meng. A New Solution Procedure for Application of Energy-Conserving Algorithms to General Constitutive Models in Nonlinear Elastodynamics. *Computer Methods in Applied Mechanics and Engineering*, 190:6309–6322, 2001.

[68] O. Gonzalez and J. C. Simo. On the Stability of Symplectic and Energy-Momentum Algorithms for Non-linear Hamiltonian Systems with Symmetry. *Computer Methods in Applied Mechanics and Engineering*, 134:197–222, 1996.

[69] O. Gonzalez. Time Integration and Discrete Hamiltonian Systems. *Journal of Nonlinear Science*, 6:449–467, 1996.

Appendix A

Nonlinear Dynamics and Example Problems

In this section, four single-degree-of-freedom example problems, i.e., the Duffing equation, Hardening spring equation, softening spring equation, and simple mathematical pendulum; and three multi-degree-of-freedom example problems, i.e., the 3-D Kepler problem, motion of a two-body problem, and tetrahedron spring-mass linear system, are briefly explained. These example problems are used to observe the characteristics of the algorithms proposed in the previous sections. Finally, extensions to hyperelastic elastodynamics with finite elements is presented.

A.1 The Duffing Nonlinear Oscillator

The Duffing oscillator is a periodically forced oscillator, and it is generally expressed as

$$\ddot{x} + c\dot{x} + \left(\beta x^3 \pm \omega_0^2 x\right) = \gamma \cos\left(\omega t + \phi\right) \tag{A.1}$$

where $c \geq 0$ is the damping constant. Eq. (A.1) is called the Duffing equation. When there is no physical damping ($c = 0$) and no applied loading ($\gamma = 0$), Eq. (A.1) becomes

$$\ddot{x} + S_1 x \left(1 + S_2 x^2\right) = 0 \tag{A.2}$$

where $S_1 = \omega_0^2 > 0$ and $S_2 = \beta/\omega_0^2$ are the stiffness constants. Consider the case of $\beta > 0$, i.e., $S_2 > 0$ (a hardening spring). Suppose the initial conditions are given: $x(t_0) = x_0$ and $\dot{x}(t_0) = \dot{x}_0$. The conservative force is given as

$$f^c = S_1 x \left(1 + S_2 x^2\right) \tag{A.3}$$

Note that the conservative force is monotonic in x. The linearized form of Eq. (A.2) can be written as

$$\ddot{x} + \nabla f^c x = 0 \tag{A.4}$$

where ∇f^c is sometimes called the load tangent stiffness, K_T, in computational mechanics, and it is given by

$$K_T = \nabla f^c = S_1 + 3S_1 S_2 x^2 \tag{A.5}$$

By integrating in time, the constant energy state is given by

$$\mathscr{E}\left(x, \dot{x}\right) = \tfrac{1}{2}\dot{x}^2 + \tfrac{1}{2}S_1 x^2 + \tfrac{1}{4}S_1 S_2 x^4 \tag{A.6}$$

Therefore, the kinetic energy and the potential energy can be written as

$$\mathscr{T}\left(\dot{x}\right) = \tfrac{1}{2}\dot{x}^2 \tag{A.7}$$

$$\mathscr{U}\left(x\right) = \tfrac{1}{4}S_1 x^2 \left(2 + S_2 x^2\right) \tag{A.8}$$

respectively.

Input parameters: The stiffness constants have the values of $S_1 = 10$ and $S_2 = 100$. The initial conditions are $q(0) = 1$ and $\dot{q}(0) = 1$. The time step size and the tolerance for the iteration are selected to be $\Delta t = 0.01$ and $\epsilon = 10^{-8}$, respectively. The initial time and final time for the simulation are $t_0 = 0$ and $T = 1$ sec, respectively.

A.2 Hardening Spring Nonlinear Oscillator

An undamped free vibration of a hardening spring equation, as illustrated in figure A.2, of unit mass with the initial conditions is described by

$$\begin{aligned} \ddot{x} + 2(S - EA)\frac{x}{\sqrt{L^2 + x^2}} + \frac{2EA}{L}x = 0 \\ x(t_0) = x_0 \ \text{ and } \ \dot{x}(t_0) = \dot{x}_0 \end{aligned} \tag{A.9}$$

where $S > 0$ is the stiffness constant, E is the Young modulus of the spring, A is the cross-section, and L is the original length. The conservative force is given as

$$f^c = 2(S - EA)\frac{x}{\sqrt{L^2 + x^2}} + \frac{2EA}{L}x \tag{A.10}$$

and the linearized form of Eq. (A.9) can be written as

$$\ddot{x} + \nabla f^c x = 0 \tag{A.11}$$

where the load tangent stiffness $K_T := \nabla f^c$ is given by

$$K_T = 2(S - EA)\frac{L^2}{(L^2 + x^2)^{3/2}} + \frac{2EA}{L} \tag{A.12}$$

By integrating in time, the constant energy state is given by

$$\mathscr{E}(x, \dot{x}) = \frac{1}{2}\dot{x}^2 + 2(S - EA)\sqrt{L^2 + x^2} + \frac{EA}{L}x^2 \tag{A.13}$$

Therefore, the kinetic energy and the potential energy can be written as

$$\mathscr{T}(\dot{x}) = \tfrac{1}{2}\dot{x}^2 \tag{A.14}$$

$$\mathscr{U}(x) = 2(S - EA)\sqrt{L^2 + x^2} + \frac{EA}{L}x^2 \tag{A.15}$$

respectively.

Input parameters: The selected input parameters for numerical simulation are: $S = 10$, $EA = 1e3$, and $L = 1$. The initial conditions are $q(0) = 1$ and $\dot{q}(0) = 1$. The time step size and the tolerance for the iteration are selected to be $\Delta t = 0.01$ and $\epsilon = 10^{-8}$, respectively. The initial time and final time for the simulation are $t_0 = 0$ and $T = 3$ sec, respectively.

A.3 Bilinear Softening Spring Nonlinear Oscillator

An undamped free vibration of a bilinear softening spring equation of unit mass with the initial conditions is described by

$$\begin{aligned} \ddot{x} + S\tanh x = 0 \\ x(t_0) = x_0 \ \text{ and } \ \dot{x}(t_0) = \dot{x}_0 \end{aligned} \tag{A.16}$$

where $S > 0$ is the stiffness constant. The conservative force is given as

$$f^c = S\tanh x \tag{A.17}$$

and the linearlized form of Eq. (A.16) can be written as

$$\ddot{x} + \nabla f^c x = 0 \tag{A.18}$$

where the load tangent stiffness $K_T := \nabla f^c$ is given by

$$K_T = S\text{sech}^2 x \tag{A.19}$$

By integrating in time, the constant energy state is given by

$$\mathscr{E}(x, \dot{x}) = \frac{1}{2}\dot{x}^2 + S\ln(\cosh x) \tag{A.20}$$

Therefore, the kinetic energy and the potential energy can be written as

$$\mathscr{T}(\dot{x}) = \tfrac{1}{2}\dot{x}^2 \tag{A.21}$$

$$\mathscr{U}(x) = S\ln(\cosh x) \tag{A.22}$$

respectively.

Input parameters: The stiffness is selected to be $S = 10$ for the numerical simulation. The initial conditions are $q(0) = 1$ and $\dot{q}(0) = 1$. The time step size and the tolerance for the iteration are selected to be $\Delta t = 0.01$ and $\epsilon = 10^{-8}$, respectively. The initial time and final time for the simulation are $t_0 = 0$ and $T = 5$ sec, respectively.

A.4 Nonlinear Oscillator [1]

Conservative Systems: Consider the autonomous Total Energy for a two-degree-of-freedom nonlinear oscillator,

$$\mathscr{E}(\mathbf{q},\dot{\mathbf{q}}) = \mathscr{T}(\dot{\mathbf{q}}) + \mathscr{U}(\mathbf{q}) \tag{A.23}$$

where the kinetic energy, $\mathscr{T}(\dot{\mathbf{q}}) : T_{\mathbf{q}}Q \to \mathbb{R}$, and the potential energy expressed in terms of $\mathbf{q}(t) : \mathbb{I} \to Q = \mathbb{R}^2$, are given by

$$\mathscr{T}(\dot{\mathbf{q}}) = \frac{1}{2}\dot{\mathbf{q}} \cdot \mathbf{M}\dot{\mathbf{q}} \text{ and } \mathscr{U}(\mathbf{q}) = \|\mathbf{q}\|^2 \left(\|\mathbf{q}\|^2 - 1\right)^2 \tag{A.24}$$

respectively. Note that the potential energy $\mathscr{U}(\mathbf{q})$ can also be written in terms of $r := \|\mathbf{q}\| = \mathbf{q} \cdot \mathbf{q} \in \mathbb{R}$ by $\mathscr{V}(r) = r^2(r^2-1)^2$. In the conservative system, $\mathscr{P}^{\text{input}} = 0$; hence, the ***principle of conservation of the Total Energy*** in the mechanical system, $d\mathscr{E}/dt = 0$, leads to the equation of motion as follows:

$$\frac{d}{dt}\left(\frac{\partial \mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial \dot{\mathbf{q}}}\right) + \frac{\partial \mathscr{E}\left(\mathbf{q},\dot{\mathbf{q}}\right)}{\partial \mathbf{q}} = \mathbf{0} \tag{A.25}$$

or

$$\mathbf{M}\ddot{\mathbf{q}} = \mathbf{f}^{\text{con}} \text{ or } \mathbf{M}\ddot{\mathbf{q}} + (6r^4 - 8r^2 + 2)\mathbf{q} = \mathbf{0} \tag{A.26}$$

where the conservative force is given as

$$\mathbf{f}^{\text{con}} = -(6r^4 - 8r^2 + 2)\mathbf{q} \tag{A.27}$$

Note that the Total Energy of the mechanical system is clearly conserved, and the symmetry of the Total Energy implies the angular momentum is also conserved in the system.

Nonconservative Dissipative Systems: Consider the same two-degree-of-freedom nonlinear oscillator given above, but now the dissipative potential is given as

$$\vartheta(\dot{\mathbf{q}}) = \frac{1}{2}C\dot{\mathbf{q}} \cdot \dot{\mathbf{q}} \tag{A.28}$$

where $C \in \mathbb{R}$ Therefore, the total power input due to the dissipative force and the time-dependent external force is given as

$$\mathscr{P}^{\text{input}} = \mathbf{f}^{\text{diss}}(\dot{\mathbf{q}}) \cdot \dot{\mathbf{q}} = -C\dot{\mathbf{q}} \cdot \dot{\mathbf{q}} \tag{A.29}$$

Hence, the ***principle of balance of the Total Energy*** in the mechanical system, $d\mathscr{E}/dt = \mathscr{P}^{\text{input}}$, leads to the equation of motion as follows:

$$\frac{d}{dt}\left(\frac{\partial \mathscr{E}\left(\mathbf{q}, \dot{\mathbf{q}}\right)}{\partial \dot{\mathbf{q}}}\right) + \frac{\partial \mathscr{E}\left(\mathbf{q}, \dot{\mathbf{q}}\right)}{\partial \mathbf{q}} = -\frac{\partial \vartheta(\mathbf{q}, \dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}} \tag{A.30}$$

or

$$\mathbf{M}\ddot{\mathbf{q}} = \mathbf{f}^{\text{con}} + \mathbf{f}^{\text{diss}}(\dot{\mathbf{q}}) \;\text{ or }\; \mathbf{M}\ddot{\mathbf{q}} + C\dot{\mathbf{q}} + (6r^4 - 8r^2 + 2)\mathbf{q} = \mathbf{0} \tag{A.31}$$

See Section 6.3 for the input information for the numerical simulation.

A.5 Simple Pendulum Equation

Consider a simple pendulum whose origin is fixed in the x-y plane and its hinge is frictionless, taking the generalized coordinate as $q(t) = \theta(t)$. The constant energy state is given by

$$\mathscr{E}\left(\theta, \dot{\theta}\right) = \tfrac{1}{2}mL^2\dot{\theta}^2 + mgL\left(1 - \cos\theta\right) \tag{A.32}$$

That is, the kinetic energy and the potential energy can be written as

$$\mathscr{T}(\dot{\theta}) = \tfrac{1}{2}mL^2\dot{\theta}^2 \tag{A.33}$$

$$\mathscr{U}\left(\theta\right) = mgL\left(1 - \cos\theta\right) \tag{A.34}$$

respectively. Notice that the generalized mass defined by mL^2 is constant. The conservative force is given as

$$f^c = mgL\sin\theta \tag{A.35}$$

and the linearized form of equation of motion may be written as

$$mL^2\ddot{\theta} + \nabla f^c\theta = 0 \tag{A.36}$$

where $\nabla f^c = \partial f^c / \partial \theta = mgL\cos\theta$.

Input parameters: The selected input parameters for the numerical simulation are: $L = 2$ and $g = 9.81$. The initial conditions are $q(0) = 1$ and $\dot{q}(0) = 1$. The time step size and the tolerance for the iteration are selected to be $\Delta t = 0.01$ and $\epsilon = 10^{-8}$, respectively. The initial time and final time for the simulation are $t_0 = 0$ and $T = 5$ sec, respectively.

A.6 Spring-Pendulum Equation

Consider the spring-pendulum problem shown in Fig. The generalized coordinates are $\mathbf{q}(t) = (\ell, \theta)^T(t) : \mathbb{I} \rightarrow Q \equiv \mathbb{R}^2$. The equation of motion is given as

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{N}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{0} \tag{A.37}$$

where the diagonal, constant mass matrix, $\mathbf{M} \in \mathbb{R}^{2\times 2}$, and the force vector, $\mathbf{N}(\mathbf{q}, \dot{\mathbf{q}}) : TQ \rightarrow \mathbb{R}^2$, take the forms

$$\mathbf{M} = \begin{bmatrix} m & 0 \\ 0 & m \end{bmatrix} \tag{A.38}$$

$$\mathbf{N}(\mathbf{q}, \dot{\mathbf{q}}) = \begin{Bmatrix} m(L_0 + \ell)\dot{\theta}^2 + mg\cos\theta - k\ell \\ -m(2\dot{r}\dot{\theta} + g\sin\theta)/(L_0 + \ell) \end{Bmatrix} \tag{A.39}$$

respectively, where m, k, L_0, and g are the mass of the pendulum, the spring constant, the length of the undeformed spring, and the gravitational acceleration, respectively.

A.7 Kepler's Problem

Consider the famous Kepler problem in $\mathbb{R}^3$ space ($N = 1$). The generalized coordinate is defined as the position of a particle of mass m, moving in the space,

and the generalized velocity is defined as its velocity:

$$\mathbf{q}(t) \equiv \mathbf{x}(t) : \mathcal{T} \to Q \equiv \mathbb{R}^3 \tag{A.40}$$

$$\dot{\mathbf{q}}(t) \equiv \dot{\mathbf{x}}(t) : \mathcal{T} \to T_q Q \equiv \mathbb{R}^3 \tag{A.41}$$

The total energy is defined as the summation of the kinetic energy and the potential energy given by

$$\mathcal{K}(\dot{\mathbf{q}}) = \frac{1}{2}\dot{\mathbf{q}} \cdot \mathbf{M}\dot{\mathbf{q}} \quad \text{and} \quad \mathcal{V}(r) = -\frac{k}{r} \tag{A.42}$$

respectively, and the constant mass matrix is defined by $\mathbf{M} = m\mathbf{I}_3$, $r = \| \mathbf{q} \| := \sqrt{\mathbf{q} \cdot \mathbf{q}}$ denotes the distance from the origin, and k is a constant. Note the potential energy satisfies $\lim_{r \to 0} \mathcal{V}(r) = 0$. Therefore, the Total Energy representation of the equation of motion

$$\frac{d}{dt}\left(\frac{\partial \mathcal{E}}{\partial \dot{\mathbf{q}}}\right) + \frac{\partial \mathcal{E}}{\partial \mathbf{q}} = \mathbf{0} \tag{A.43}$$

leads to

$$\mathbf{M}\ddot{\mathbf{q}} + \frac{k}{r^2}\frac{\mathbf{q}}{r} = \mathbf{0} \tag{A.44}$$

See Section 6.3 for the input information for the numerical simulation.

A.8 Lennard-Jones (5, 3) Potential 2-body Problem

Consider a system of two particles as shown in Fig. A.8. We have purposely selected this model problem to carefully illustrate the conservation properties, i.e., total linear and angular momenta and total energy of a system. This model problem in $\mathbb{R}^2$ space has been given in Betsch and Steinmann [53]. Define the generalized coordinate and velocity as

$$\mathbf{q}(t) = (\mathbf{q}^1(t), \mathbf{q}^2(t)) \equiv \mathbf{x} : \mathbb{I} \to Q \equiv \mathbb{R}^6 \tag{A.45}$$

$$\dot{\mathbf{q}}(t) = (\dot{\mathbf{q}}^1(t), \dot{\mathbf{q}}^2(t)) \equiv \dot{\mathbf{x}} : \mathbb{I} \to TQ \equiv \mathbb{R}^6 \tag{A.46}$$

respectively. The total energy of the system is defined by

$$\mathscr{E}(\mathbf{q},\dot{\mathbf{q}}) = \mathscr{T}(\dot{\mathbf{q}}) + \mathscr{V}^{\text{int}}(r) \tag{A.47}$$

where the kinetic energy and the internal potential energy are given by

$$\mathscr{T}(\dot{\mathbf{q}}) = \frac{1}{2}\dot{\mathbf{q}}\cdot\mathbf{M}\dot{\mathbf{q}} \quad \text{and} \quad \mathscr{V}^{\text{int}}(r) = \alpha\left[\left(\frac{\sigma}{r}\right)^5 - \left(\frac{\sigma}{r}\right)^3\right] \tag{A.48}$$

respectively, where the constant mass matrix is defined by $\mathbf{M} = \text{diag}(\mathbf{m}_1, \mathbf{m}_2)$ for $\mathbf{m}_A = m_A\mathbf{I}_3$ $(A = 1, 2)$; $r = r_{12} = r_{21} := \parallel \mathbf{q}^1 - \mathbf{q}^2 \parallel$ is the inter-particle distance, and α and σ are constant parameters. Therefore, the total energy equation of motion is

$$\frac{d}{dt}\left(\frac{\partial\mathscr{E}(\mathbf{q},\dot{\mathbf{q}})}{\partial\dot{\mathbf{q}}}\right) + \frac{\partial\mathscr{E}(\mathbf{q},\dot{\mathbf{q}})}{\partial\mathbf{q}} = \mathbf{0} \text{ leads to } m_A\ddot{\mathbf{q}}^A_{n+1} + \sum_{B\neq A}^{N=2}\frac{d\mathscr{V}^{\text{int}}(r)}{dr}\frac{\mathbf{q}^A - \mathbf{q}^B}{r} = \mathbf{0} \tag{A.49}$$

where the magnitude of the internal force is given by

$$\frac{d\mathscr{V}^{\text{int}}(r)}{dr} = \frac{\alpha}{\sigma}\left[3\left(\frac{\sigma}{r}\right)^4 - 5\left(\frac{\sigma}{r}\right)^6\right] \tag{A.50}$$

See Section 6.3 for the input information for the numerical simulation.

Illustration of Residual vectors and Jacobian matrices

The residuals of the algorithms applied for the nonlinear oscillator problem, for instance, in the conservative system are shown below as an illustration. The residual for the algorithm with Options I is defined by

$$\mathbf{r}_{\text{I}} := \mathbf{M}\tilde{\mathbf{a}} - \tilde{\mathbf{f}}^{\text{con}} \equiv \mathbf{0} \tag{A.51}$$

where the algorithmic conservative force vector is given by

$$\tilde{\mathbf{f}}^{\text{con}} = \mathbf{f}^{\text{con}}_n + W_3\Delta\mathbf{f}^{\text{con}} + (W_1 - W_3)\dot{\mathbf{f}}^{\text{con}}_n\Delta t \tag{A.52}$$

The residual for the algorithm with Options II is defined by

$$\mathbf{r}_{\mathrm{II}} := \mathbf{M}\tilde{\mathbf{a}} + (6r(\tilde{\mathbf{q}})^4 - 8r(\tilde{\mathbf{q}})^2 + 2)\tilde{\mathbf{q}} \equiv \mathbf{0} \tag{A.53}$$

where the algorithmic configuration vector is given by

$$\tilde{\mathbf{q}} = \mathbf{q}_n + W_1\Lambda_1\dot{\mathbf{q}}_n\Delta t + W_2\Lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + W_3\Lambda_3\eta_2\Delta\ddot{\mathbf{q}}\Delta t^2 \tag{A.54}$$

The residual for the algorithm with Options III is defined by

$$\mathbf{r}_{\mathrm{III}} := \mathbf{M}\tilde{\mathbf{a}} + (6\tilde{r}^4 - 8\tilde{r}^2 + 2)\tilde{\mathbf{q}} \equiv \mathbf{0} \tag{A.55}$$

where the algorithmic configuration vector is given by Eq. (A.54) and the algorithmic inter-particle distance is given by

$$\tilde{r} = r_n + W_3\Delta r + (W_1 - W_3)\dot{r}_n\Delta t \tag{A.56}$$

The Jacobian matrices for Options I, II, and III are defined by

$$\mathbf{J} := \frac{\partial \mathbf{r}_i}{\partial \mathbf{q}} \tag{A.57}$$

for $i = \mathrm{I}, \mathrm{II}, \mathrm{III}$. The updates are given by

$$\begin{aligned}
\mathbf{q}_{n+1} &= \mathbf{q}_n + \lambda_1\dot{\mathbf{q}}_n\Delta t + \lambda_2\ddot{\mathbf{q}}_n\Delta t^2 + \lambda_3\Delta\mathbf{a}\Delta t^2\eta_3 \\
\dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + \lambda_4\ddot{\mathbf{q}}_n\Delta t + \lambda_5\Delta\mathbf{a}\Delta t \\
\ddot{\mathbf{q}}_{n+1} &= \ddot{\mathbf{q}}_n + \Delta\mathbf{a}
\end{aligned} \tag{A.58}$$

for all options.

A.9 Numerical Results for the SDOF Nonlinear Problems

Numerical results for the Duffing oscillator equation, hardening spring equation, bilinear softening spring equation, and simple pendulum, using the implicit GSSSS

family of algorithms and the PCE GSSSS ($\eta_3 = 0$) family of algorithms are shown below. The input parameters are listed in Table A.1. The selected algorithms are the following; see the captions of Fig. A.3 - Fig. A.65.

Implicit GSSSS family of algorithms (Option I): Algorithms 15 and 16

Implicit GSSSS family of algorithms (Option II): Algorithms 17 and 18

Implicit GSSSS family of algorithms (Option III): Algorithm 22

PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option I): Algorithm 28 with $\eta_3 = 0$

PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option II): Algorithm 29 with $\eta_3 = 0$

PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option III): Algorithm 30 with $\eta_3 = 0$

For each algorithm, we select,

U0V0/V0U0(ρ_∞,1.0,ρ_∞)

U0V0(1.0,1.0,ρ_∞^{s})

V0U0(1.0,1.0,ρ_∞^{s})

where

$$\begin{aligned} &\rho_\infty = \rho_\infty^{\min} = \rho_\infty^{\max} \in \{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\} \\ &\rho_\infty^{\mathrm{s}} \in \{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\} \end{aligned} \tag{A.59}$$

The legends of the figures are shown in Fig. 4.5. The time histories of the configuration, velocity, acceleration, and the mechanical system energy for the SDOF problems described are shown in Fig. A.3 - Fig. A.20 (Duffing oscillator), Fig. A.21 - Fig. A.35 (hardening spring), Fig. A.36 - Fig. A.50 (softening spring), and Fig. A.51 - Fig. A.65 (simple pendulum). The results of some of the other examples are shown in the text previously. The mechanical energy of the system in $[t_n, t_{n+1}]$ is not conserved exactly for the explicit schemes. For the implicit schemes, the mechanical energy is exactly conserved if we define a scalar $\ell > 0$ which makes the potential $\mathscr{U} \equiv \mathscr{V}(\ell)$ quadratic in $\ell \in \mathbb{R}_+$. For the Duffing oscillator equation, for example, if we define $\ell = q^2 = x^2$, we have $\mathscr{V} = \frac{1}{2}S_1\ell + \frac{1}{4}S_1S_2\ell^2$; see Fig. A.9 - Fig. A.11. The mechanical energy is always bounded for the symplectic schemes such as some particular families within Algorithms 17 and 18 (implicit) and Algorithms 28, 29, and 30 ($\eta_3 = 0$) (explicit).

Problems	Input parameters
Duffing oscillator	$S_1 = 10$, $S_2 = 100$
Hardening spring	$S = 10$, $EA = 1e3$, $L = 1$
Softening spring	$S = 10$
Simple pendulum	$L = 2$, $g = 9.81$

Table A.1: Input parameters for the SDOF problems

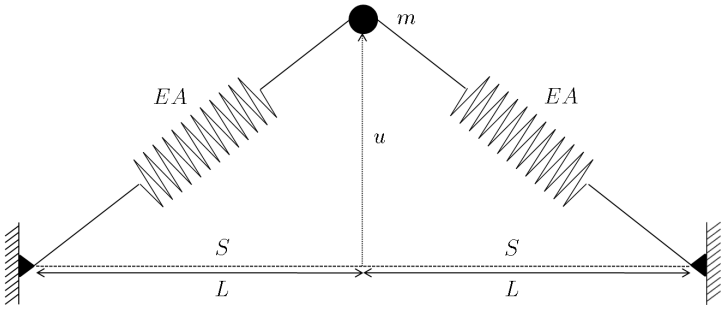


Figure A.1: The problem description of the hardening spring problem

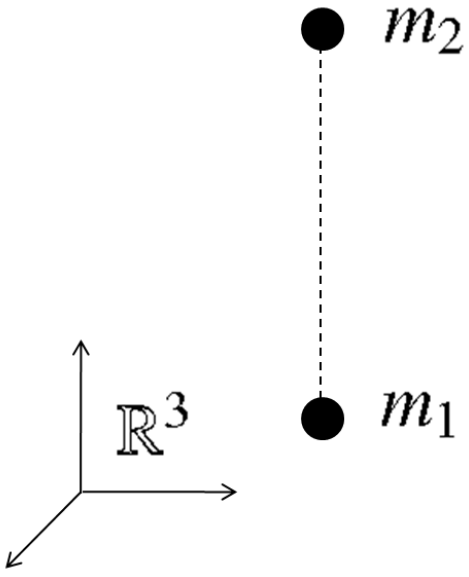
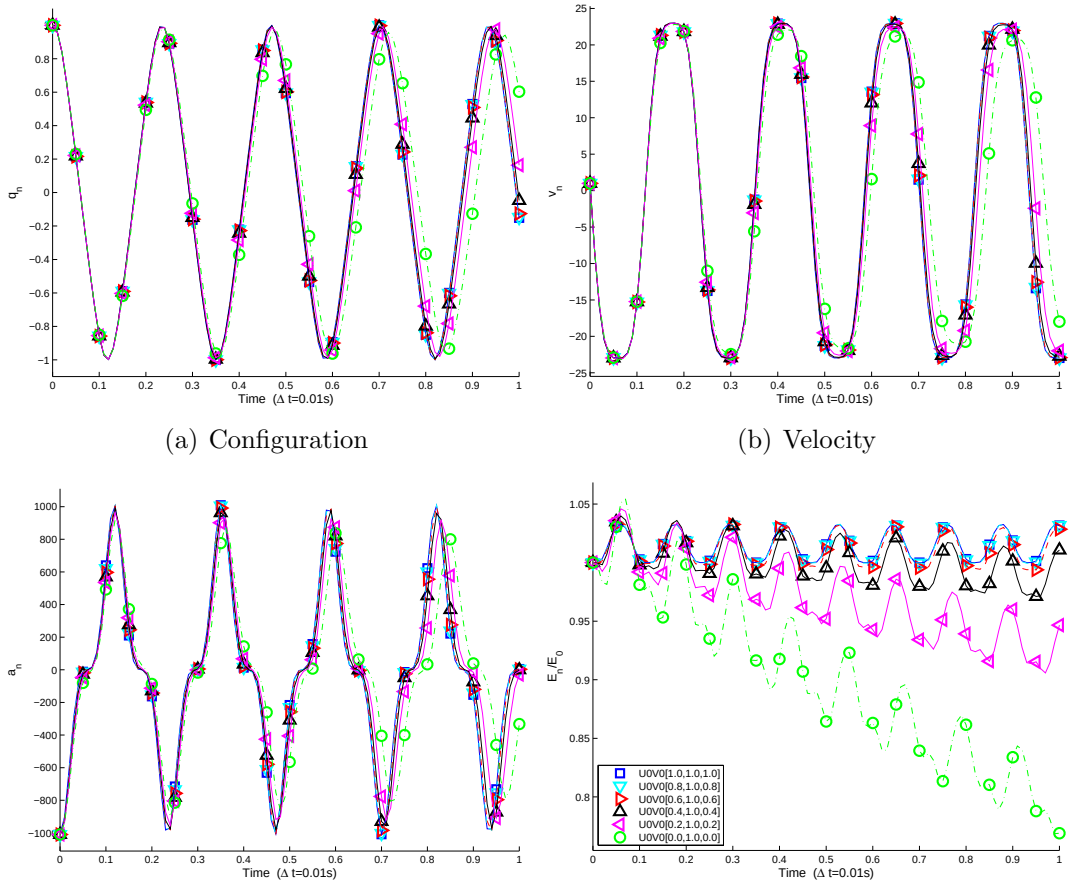


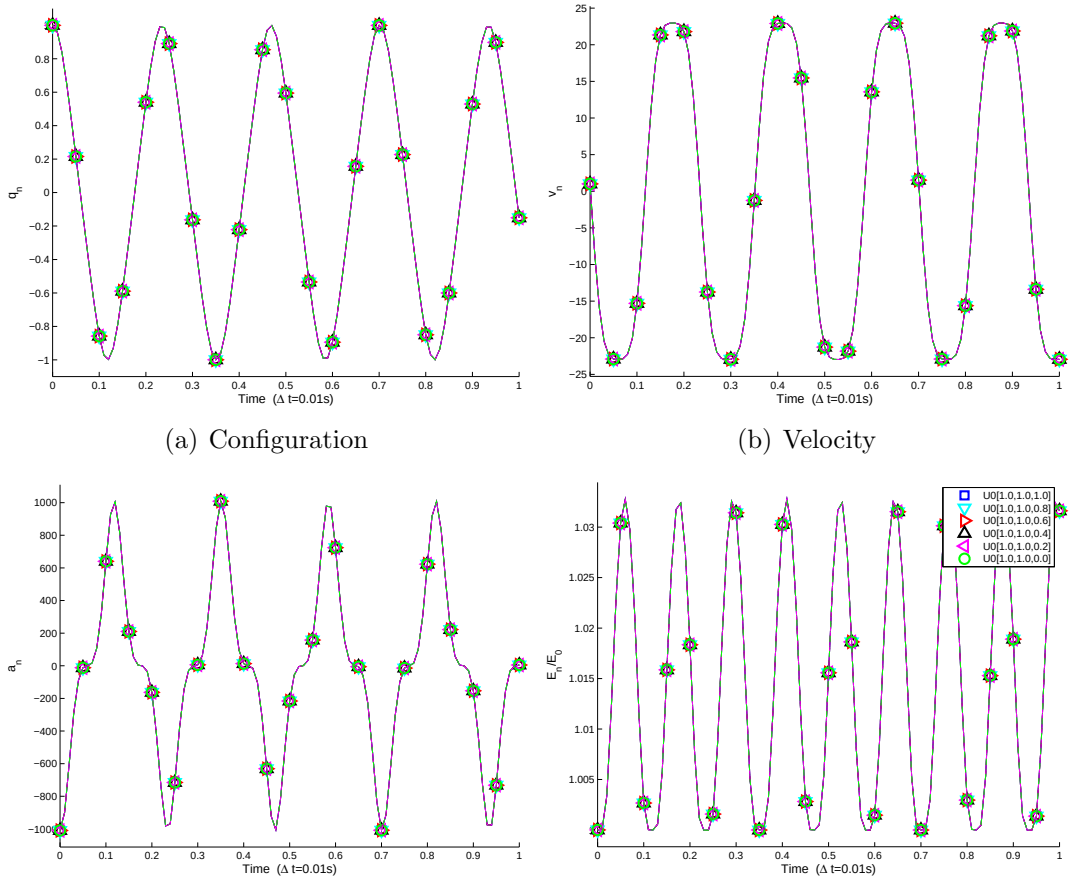
Figure A.2: The problem description of the 2-body problem



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

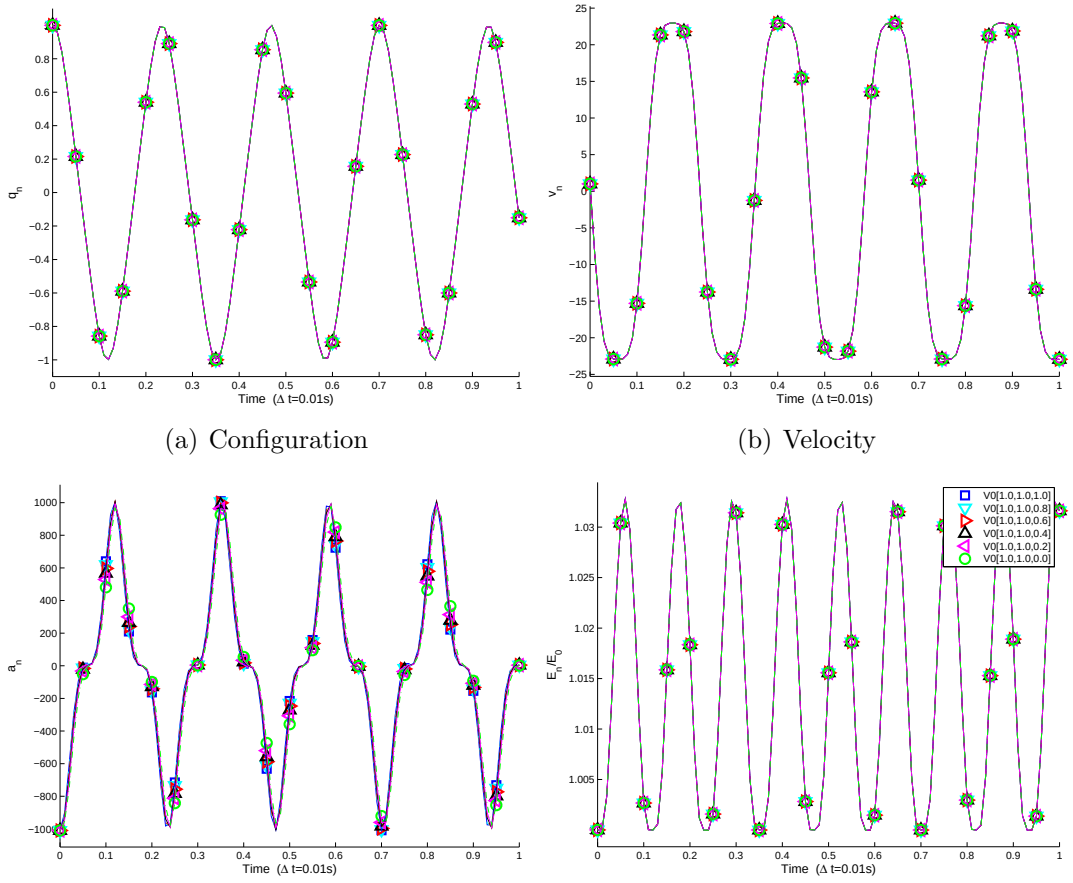
Figure A.3: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Duffing oscillator equation] [Implicit GSSSS family of algorithms (Option I) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

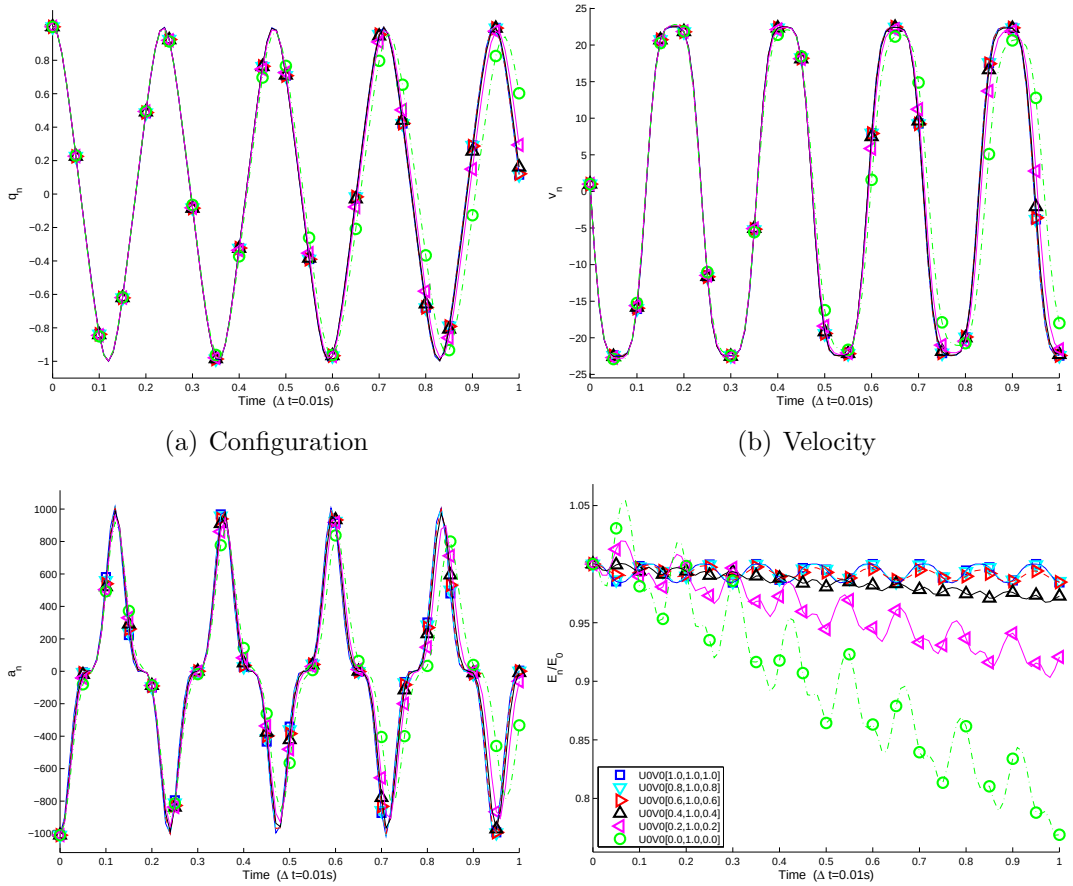
Figure A.4: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Duffing oscillator equation] [Implicit GSSSS family of algorithms (Option I) - U0V0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

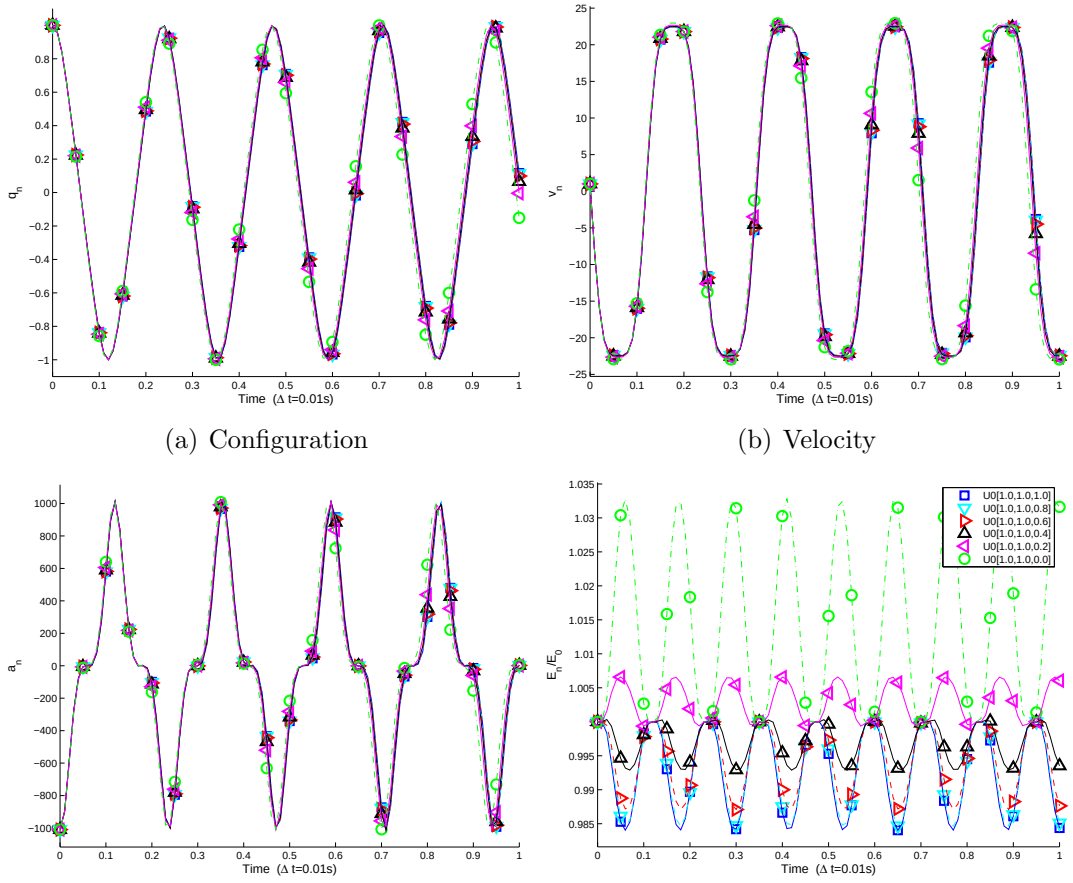
Figure A.5: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Duffing oscillator equation] [Implicit GSSSS family of algorithms (Option I) - V0U0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

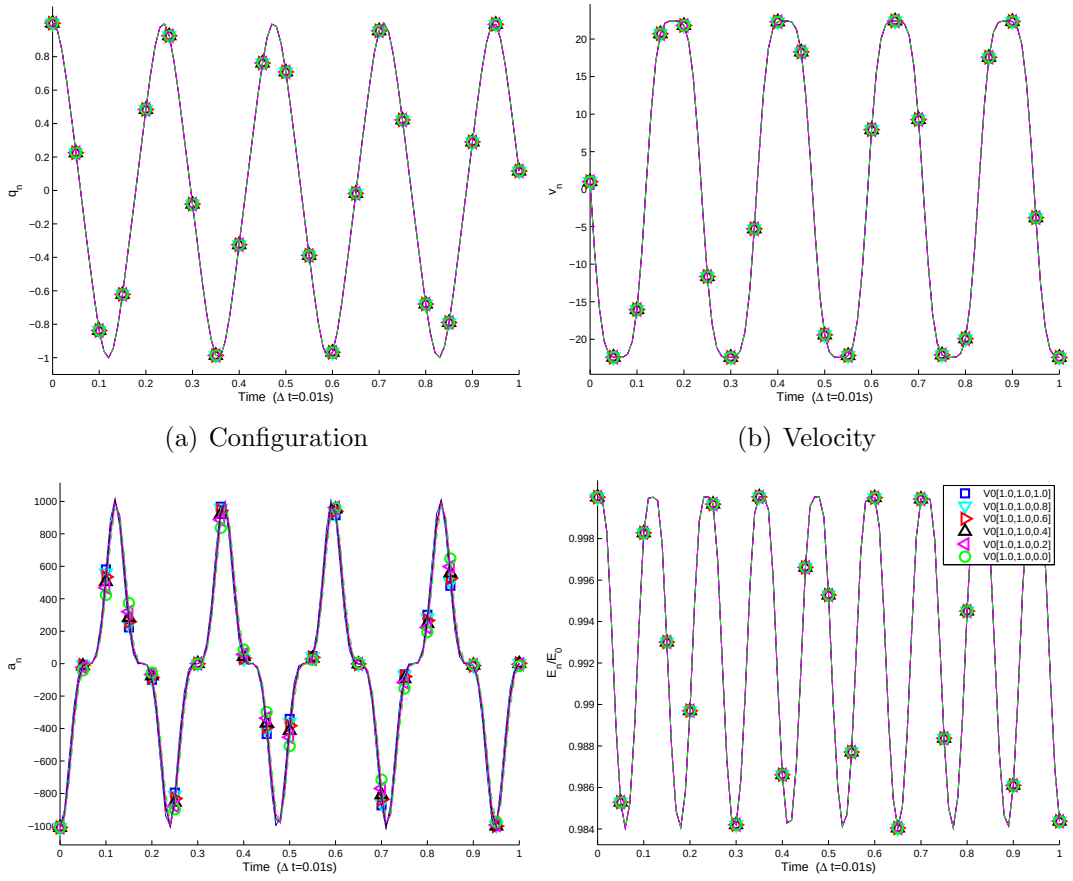
Figure A.6: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Duffing oscillator equation] [Implicit GSSSS family of algorithms (Option II) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

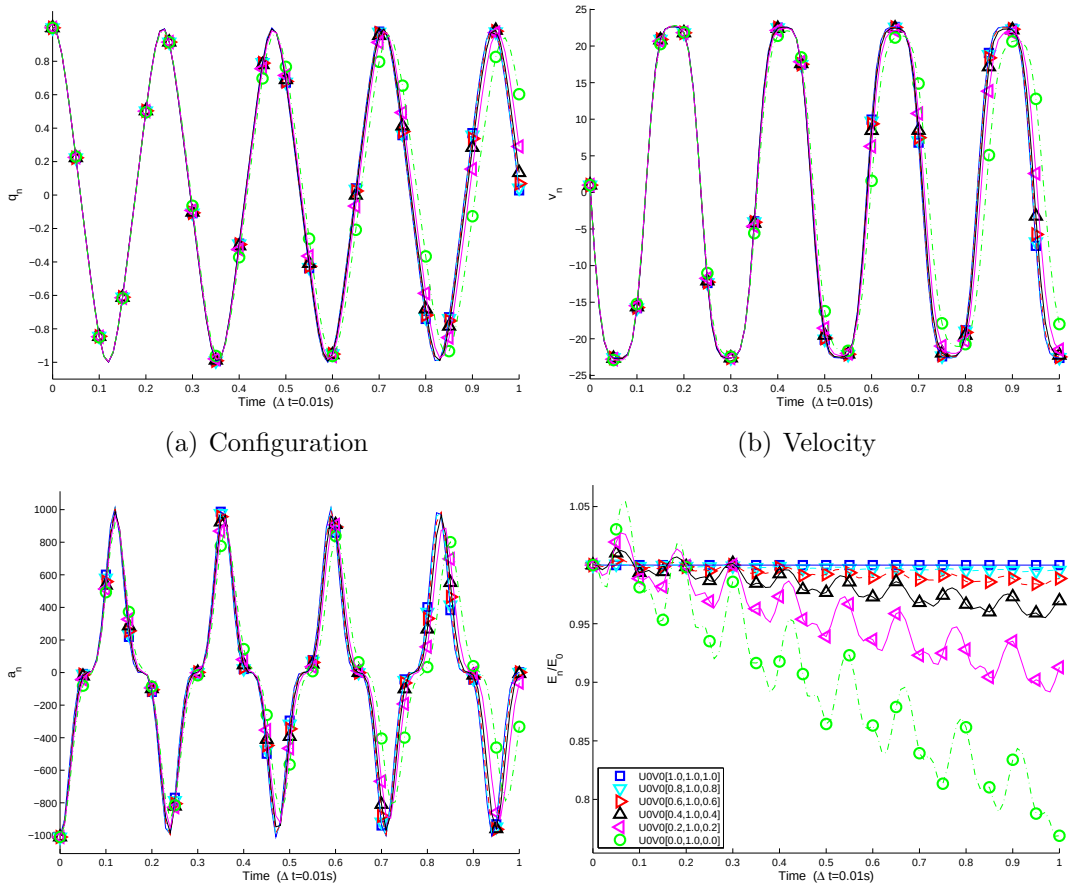
Figure A.7: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Duffing oscillator equation] [Implicit GSSSS family of algorithms (Option II) - U0V0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

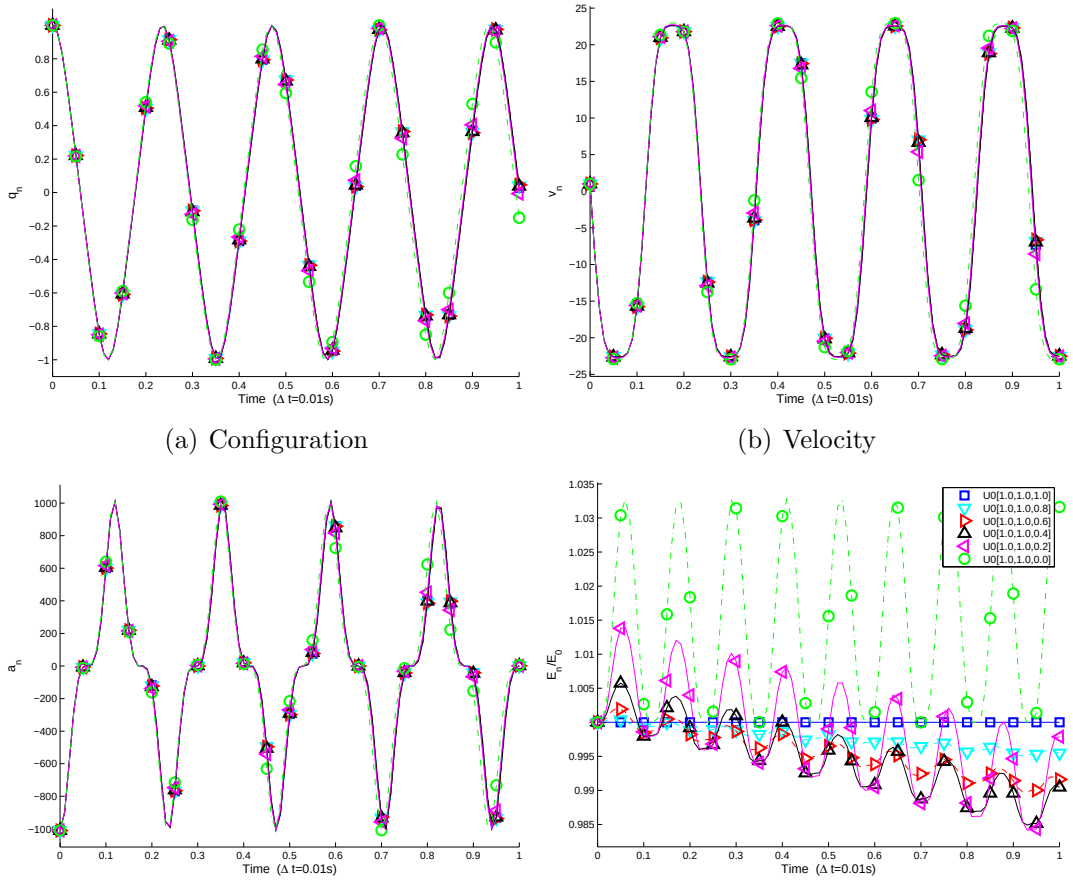
Figure A.8: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Duffing oscillator equation] [Implicit GSSSS family of algorithms (Option II) - V0U0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

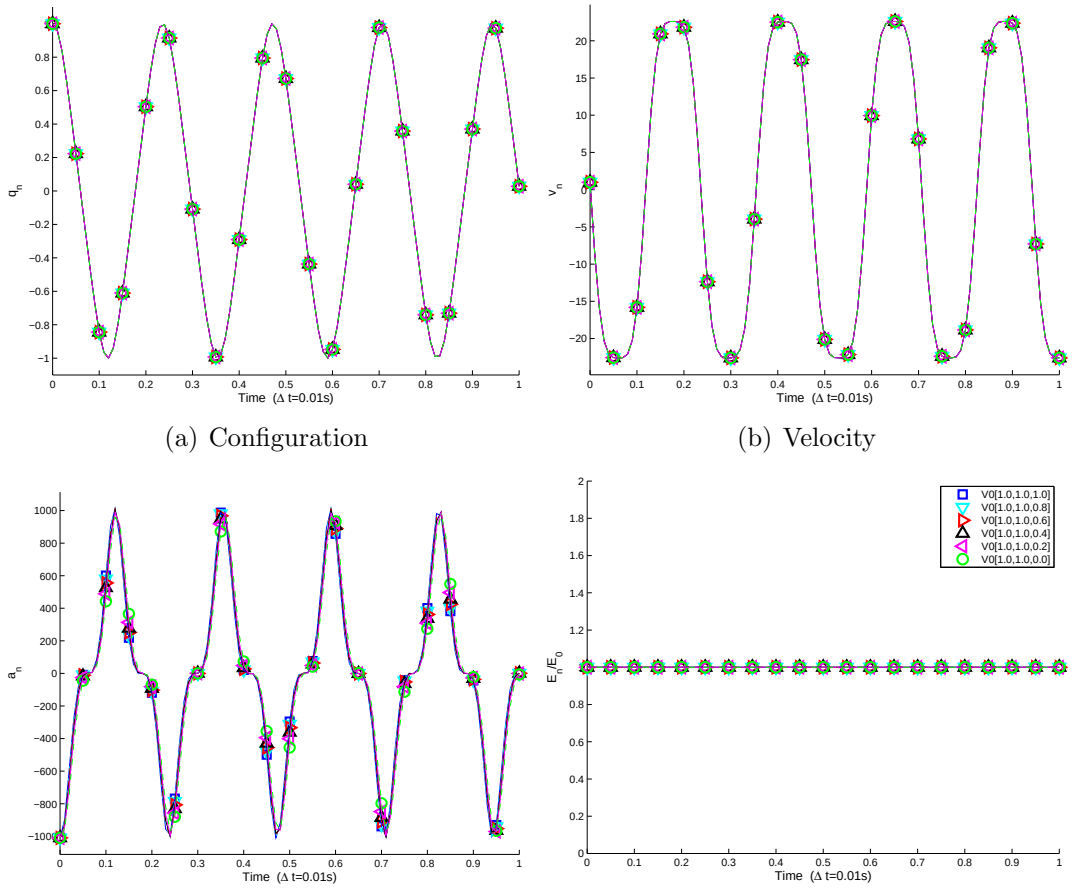
Figure A.9: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Duffing oscillator equation] [Implicit GSSSS family of algorithms (Option III) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

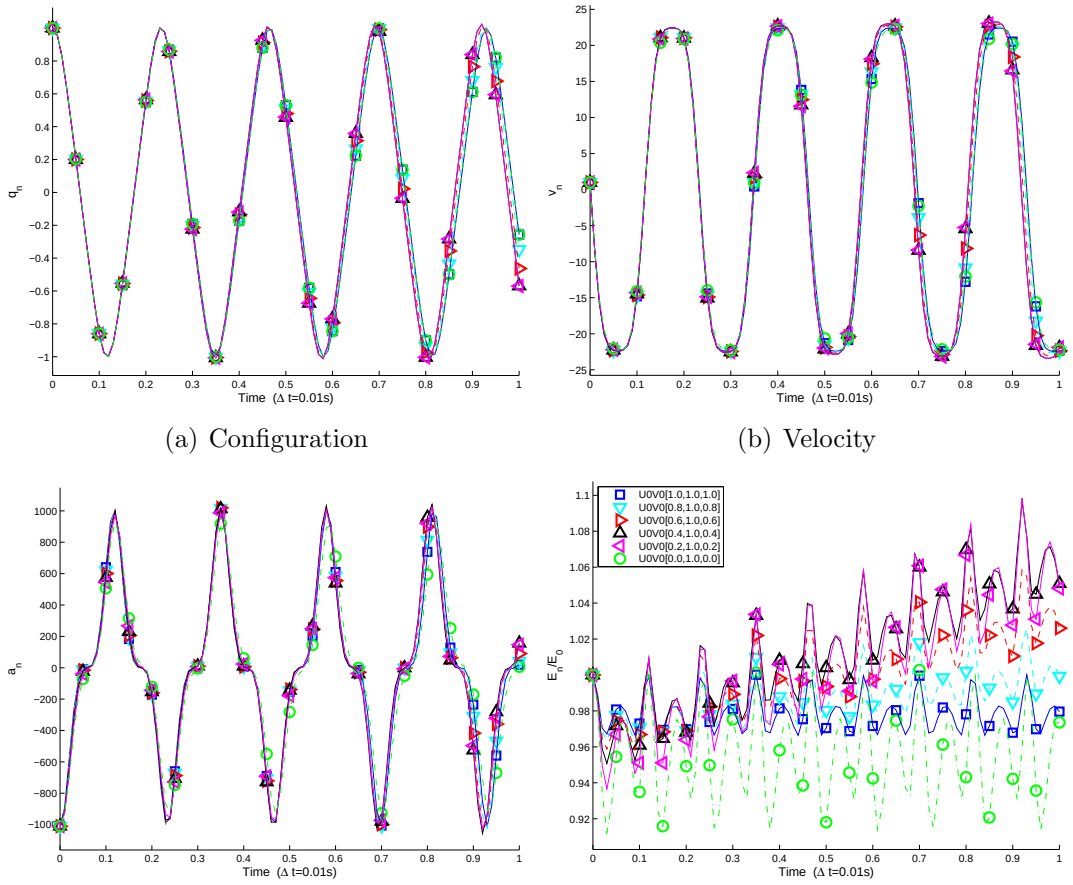
Figure A.10: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Duffing oscillator equation] [Implicit GSSSS family of algorithms (Option III) - U0V0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

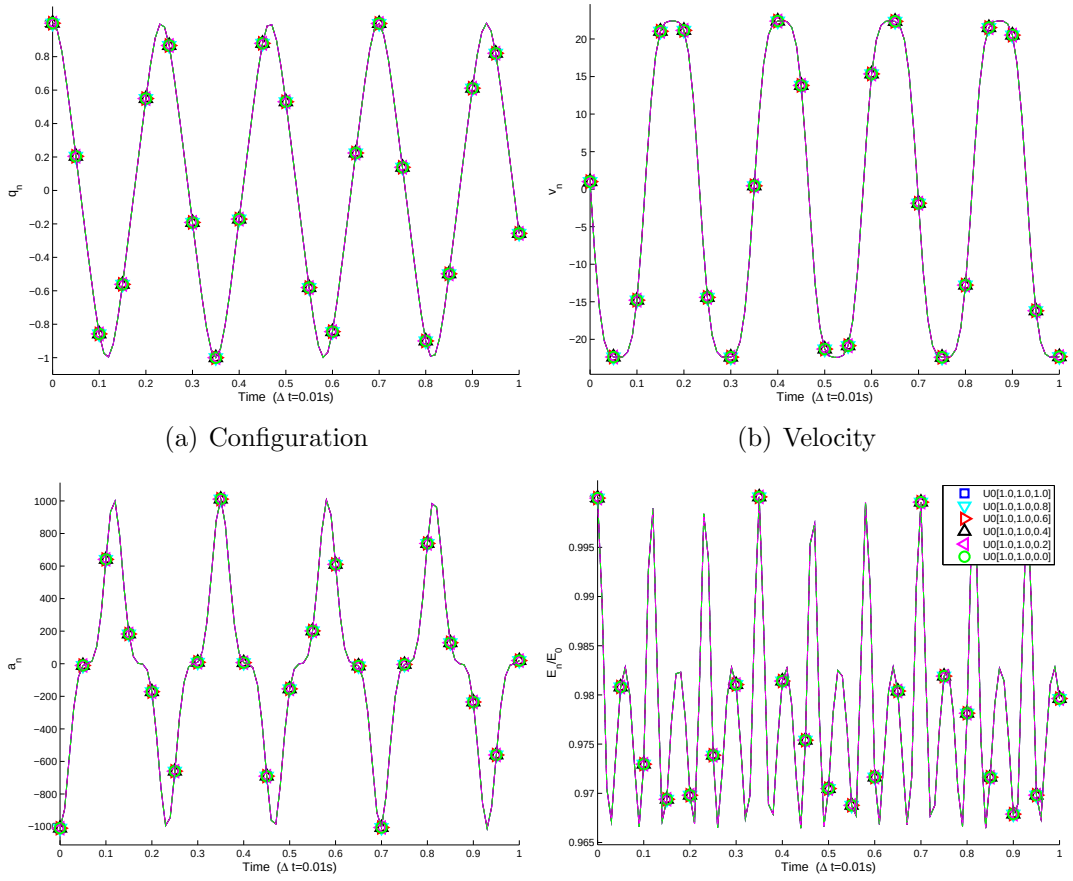
Figure A.11: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Duffing oscillator equation] [Implicit GSSSS family of algorithms (Option III) - V0U0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

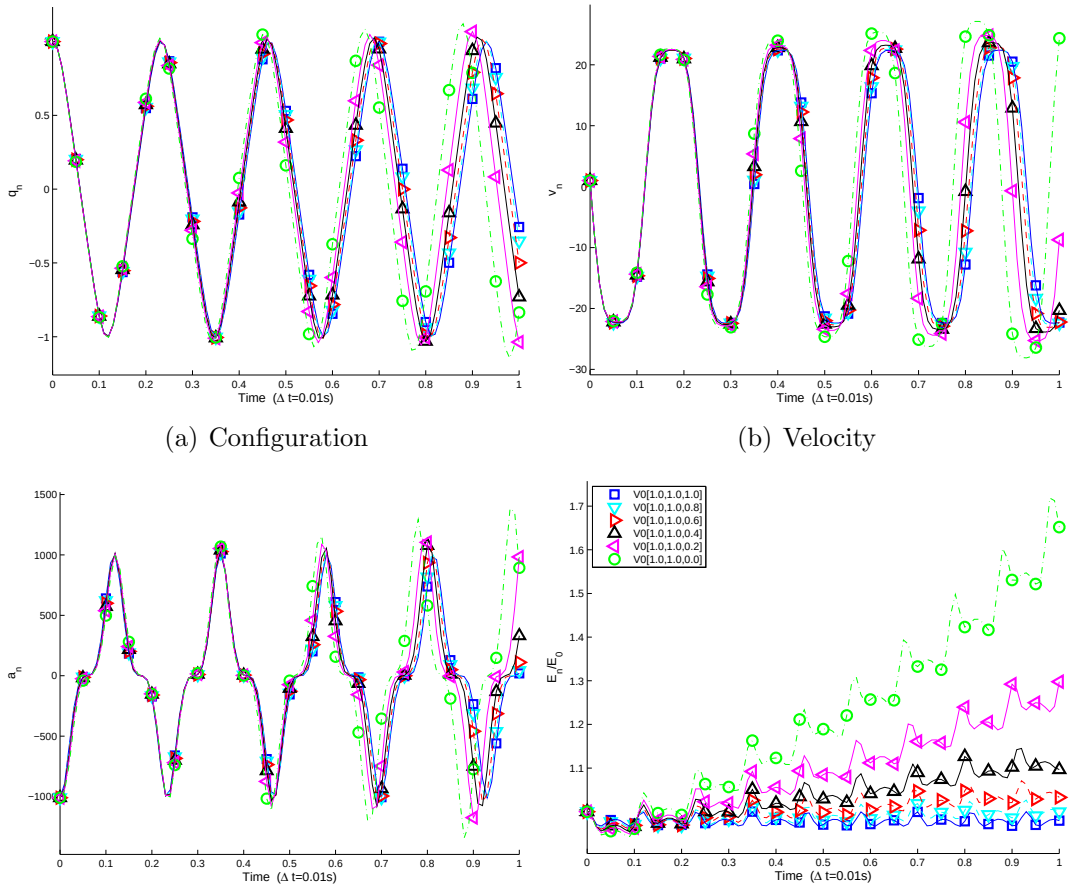
Figure A.12: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Duffing oscillator equation] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option I) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

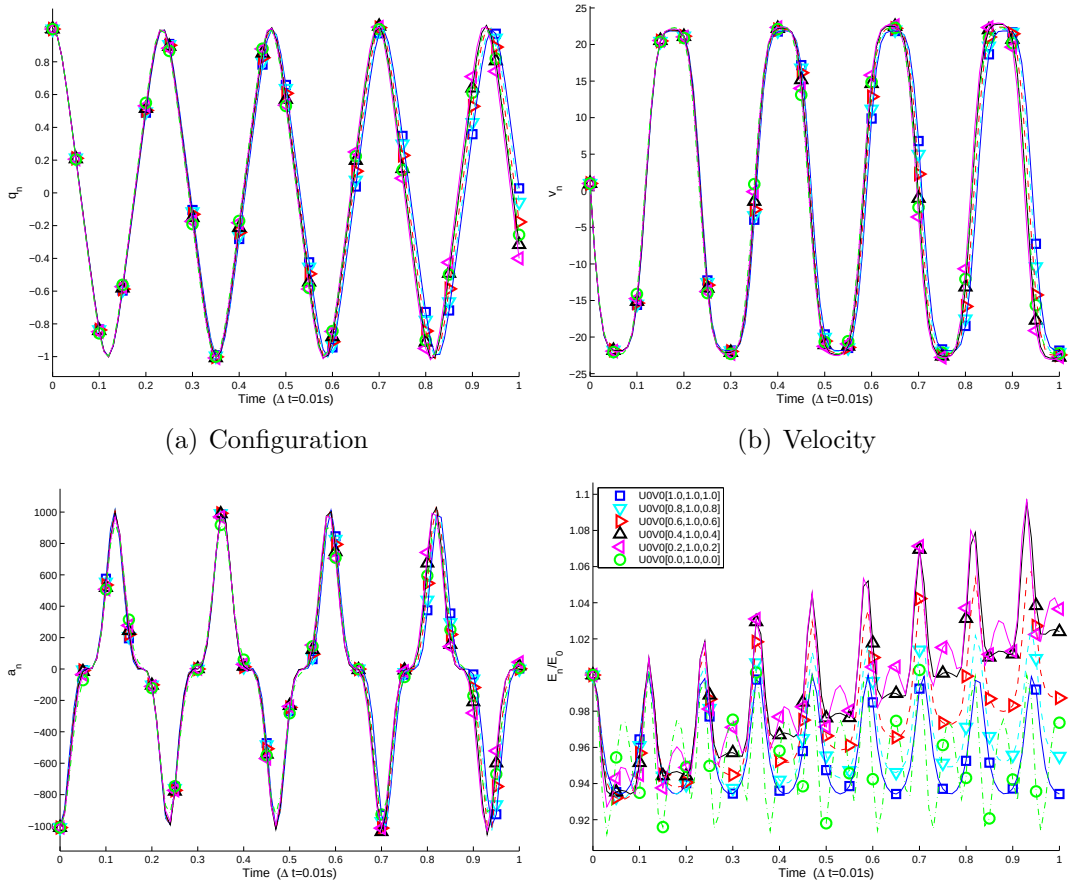
Figure A.13: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Duffing oscillator equation] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option I) - U0V0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

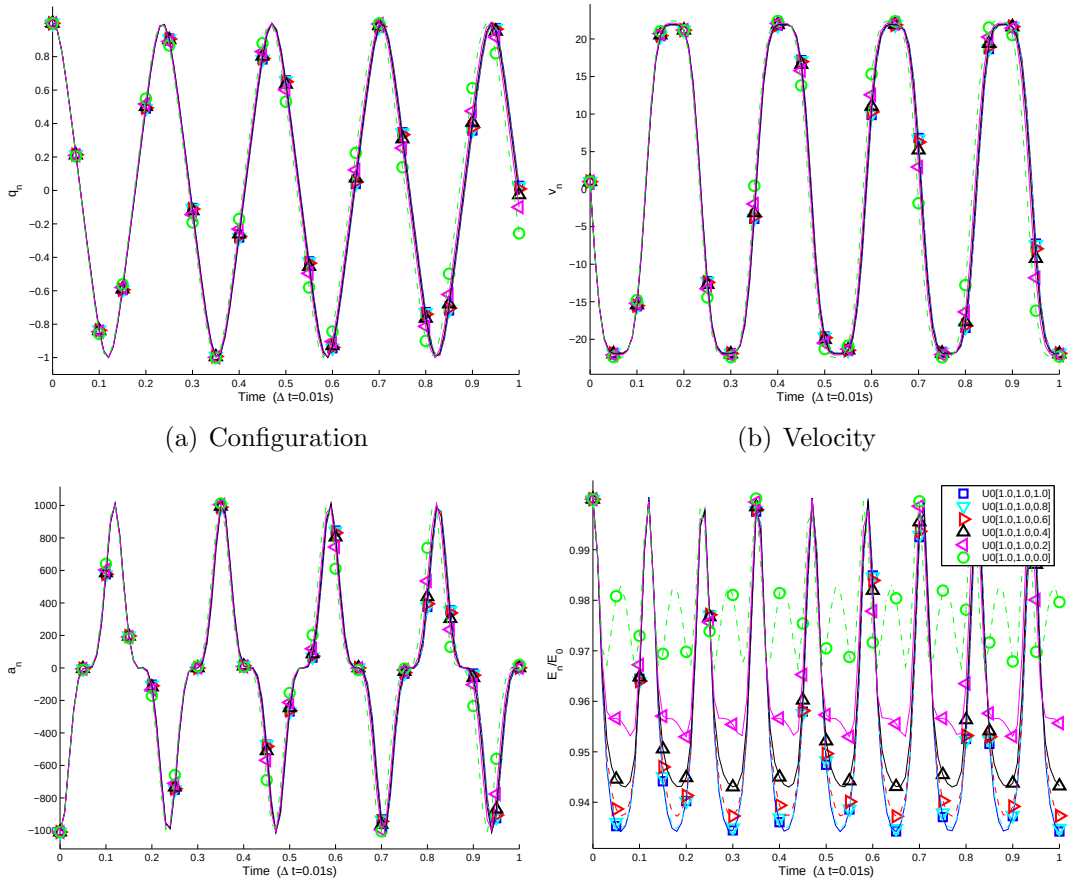
Figure A.14: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Duffing oscillator equation] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option I) - V0U0(1.0,1.0,$\rho^{\mathrm{s}}_{\infty}$)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

Figure A.15: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Duffing oscillator equation] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option II) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



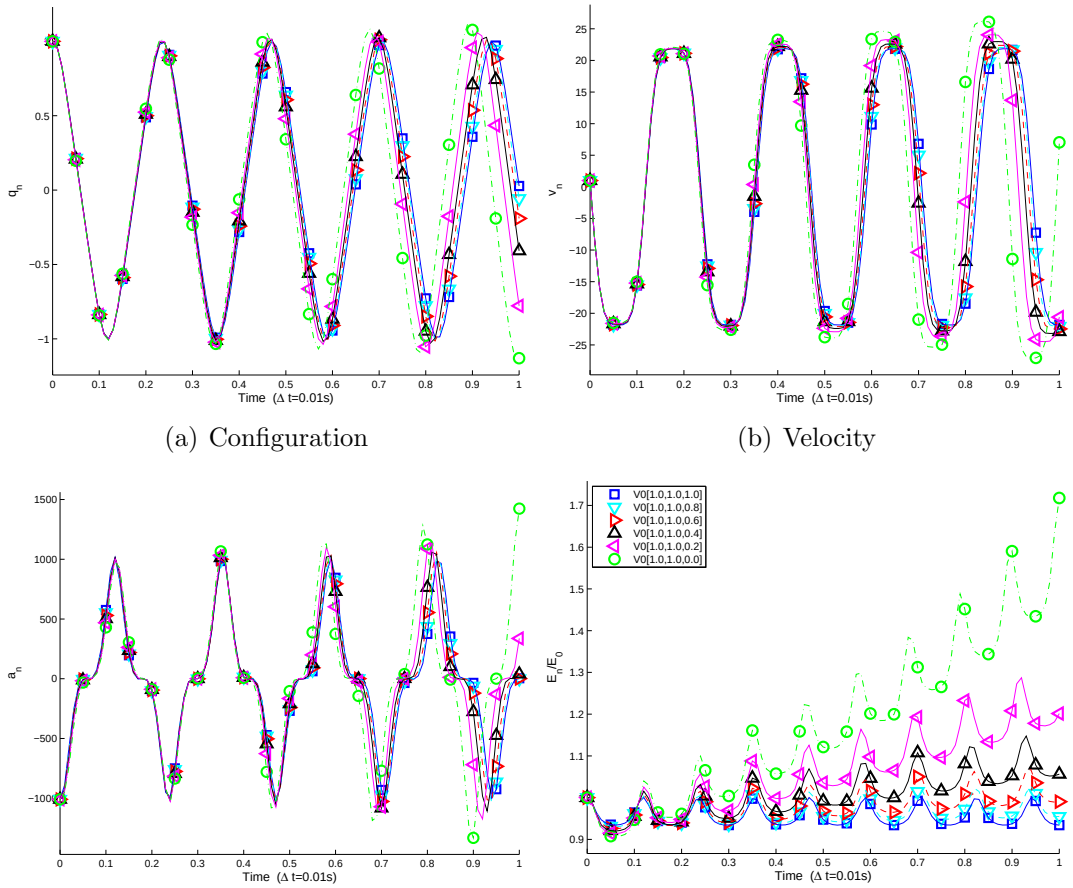
(a) Configuration

(b) Velocity

(c) Acceleration

(d) Total Energy Ratio

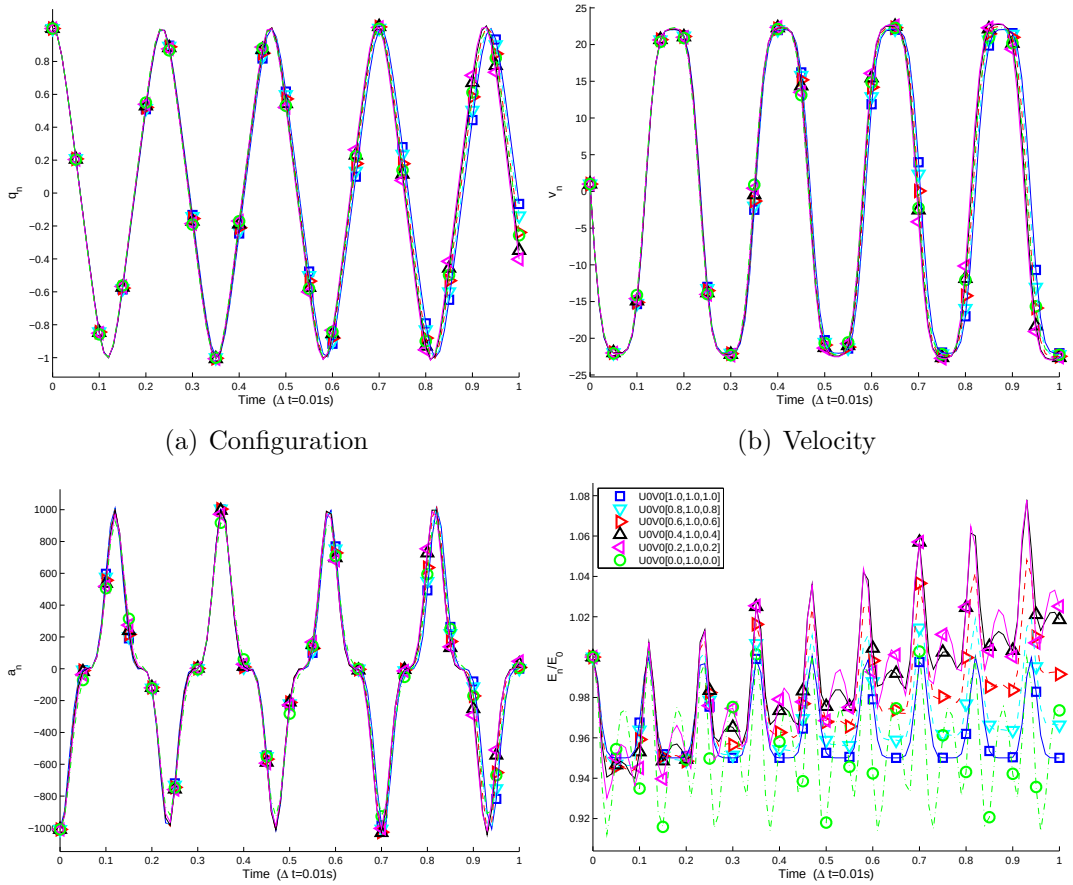
Figure A.16: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Duffing oscillator equation] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option II) - U0V0(1.0,1.0,$\rho^{\mathrm{s}}_{\infty}$)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

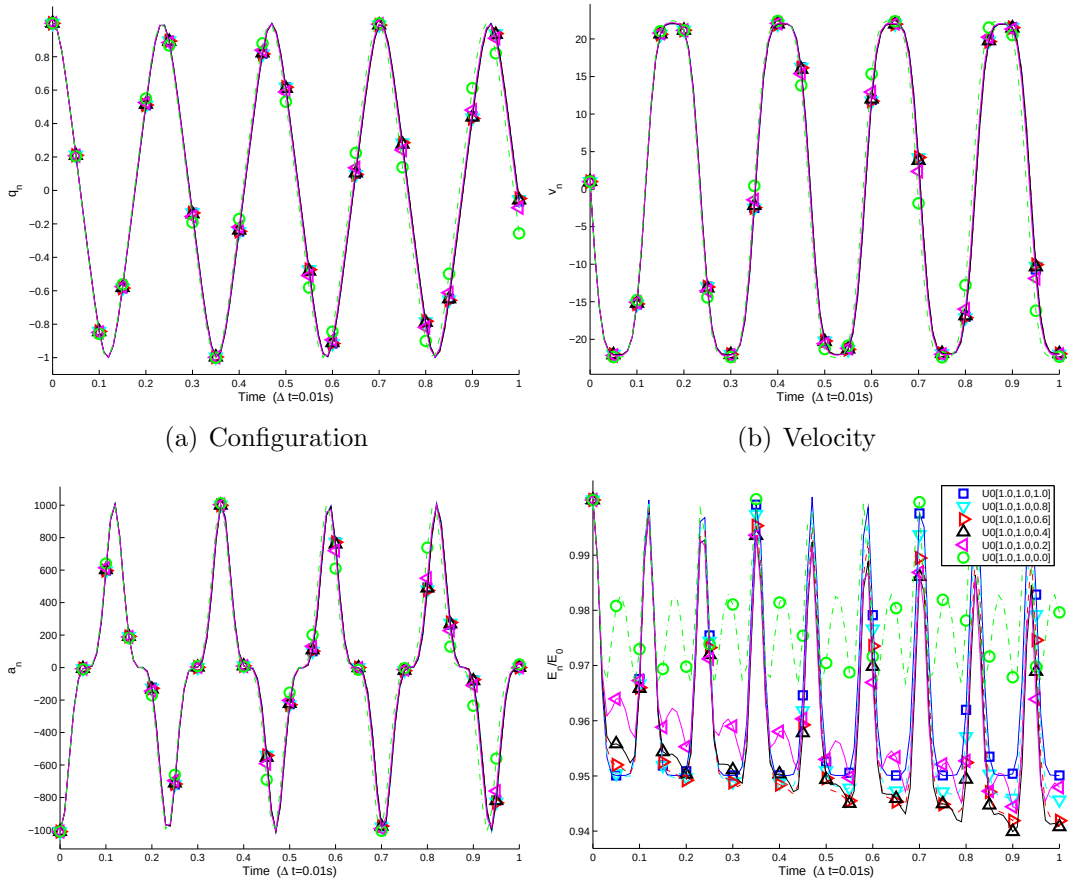
Figure A.17: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Duffing oscillator equation] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option II) - V0U0(1.0,1.0,$\rho^{\mathrm{s}}_{\infty}$)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

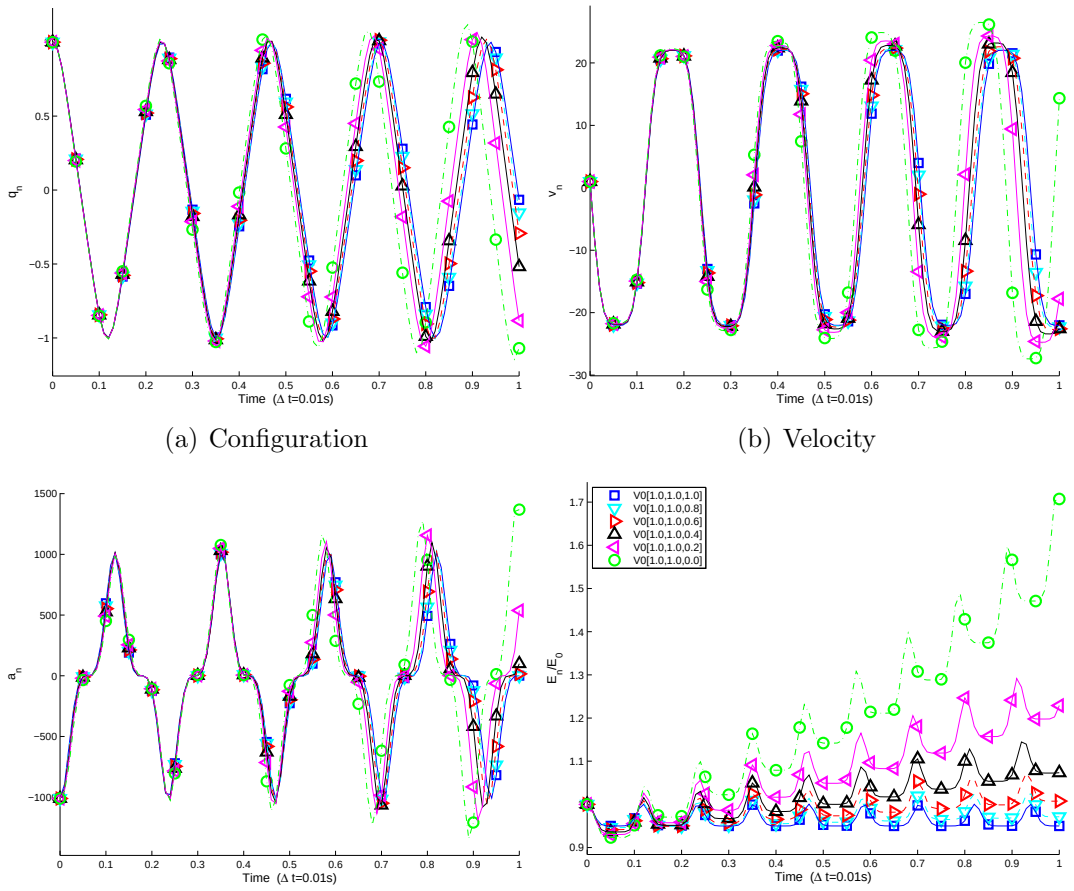
Figure A.18: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Duffing oscillator equation] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option III) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

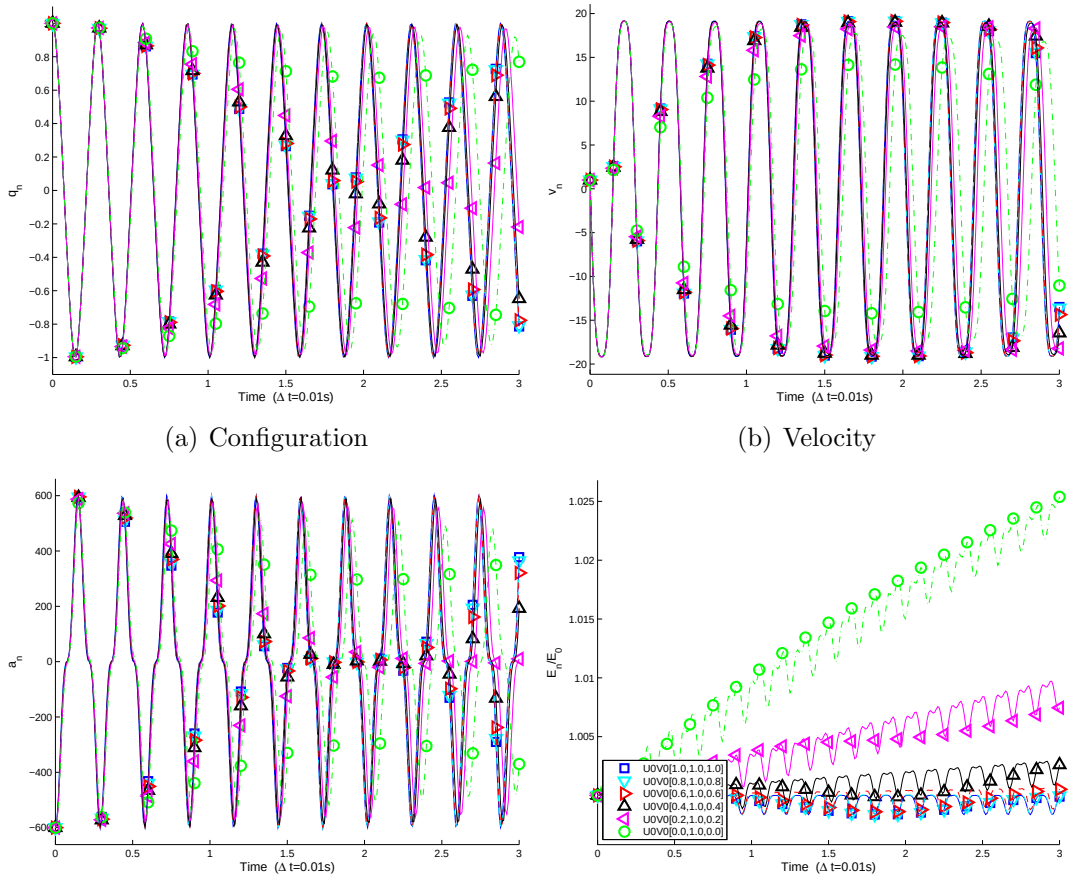
Figure A.19: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Duffing oscillator equation] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option III) - U0V0(1.0,1.0,$\rho^{\mathrm{s}}_{\infty}$)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

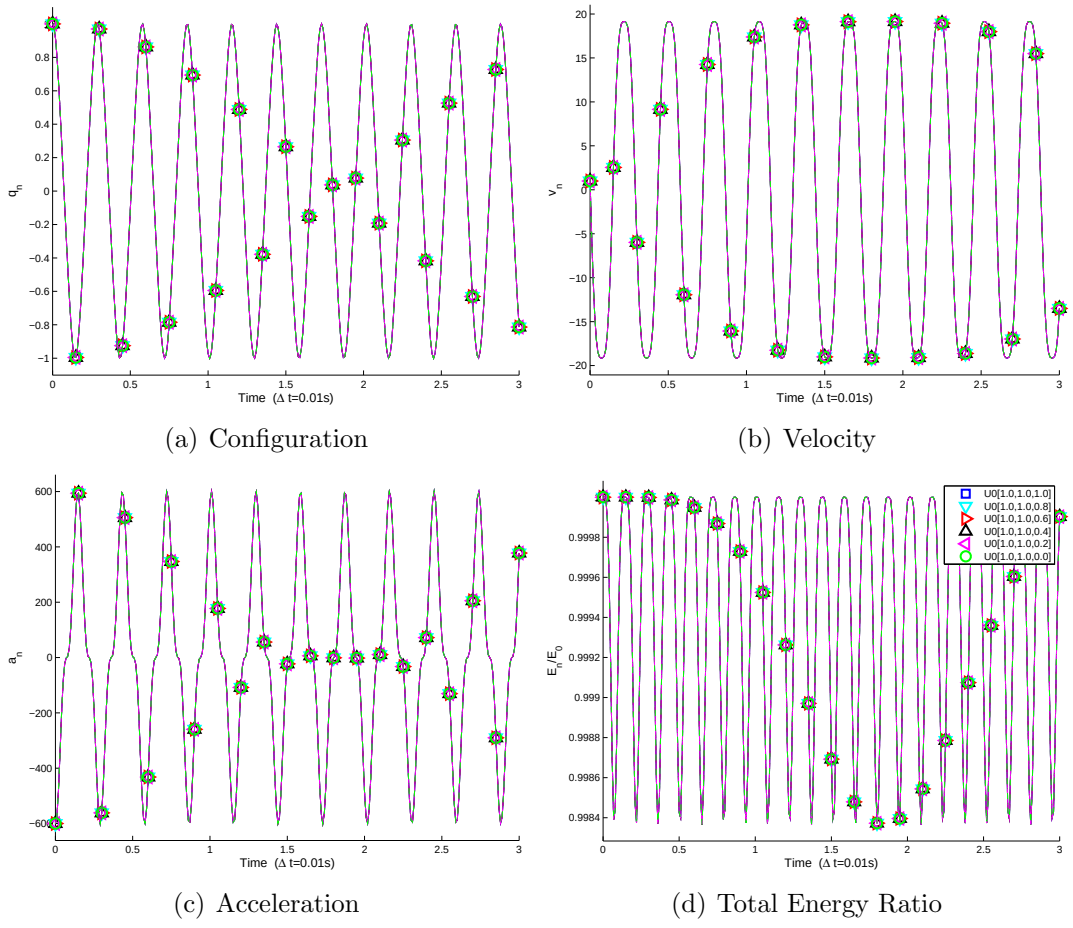
Figure A.20: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Duffing oscillator equation] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option III) - V0U0(1.0,1.0,ρ^s_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

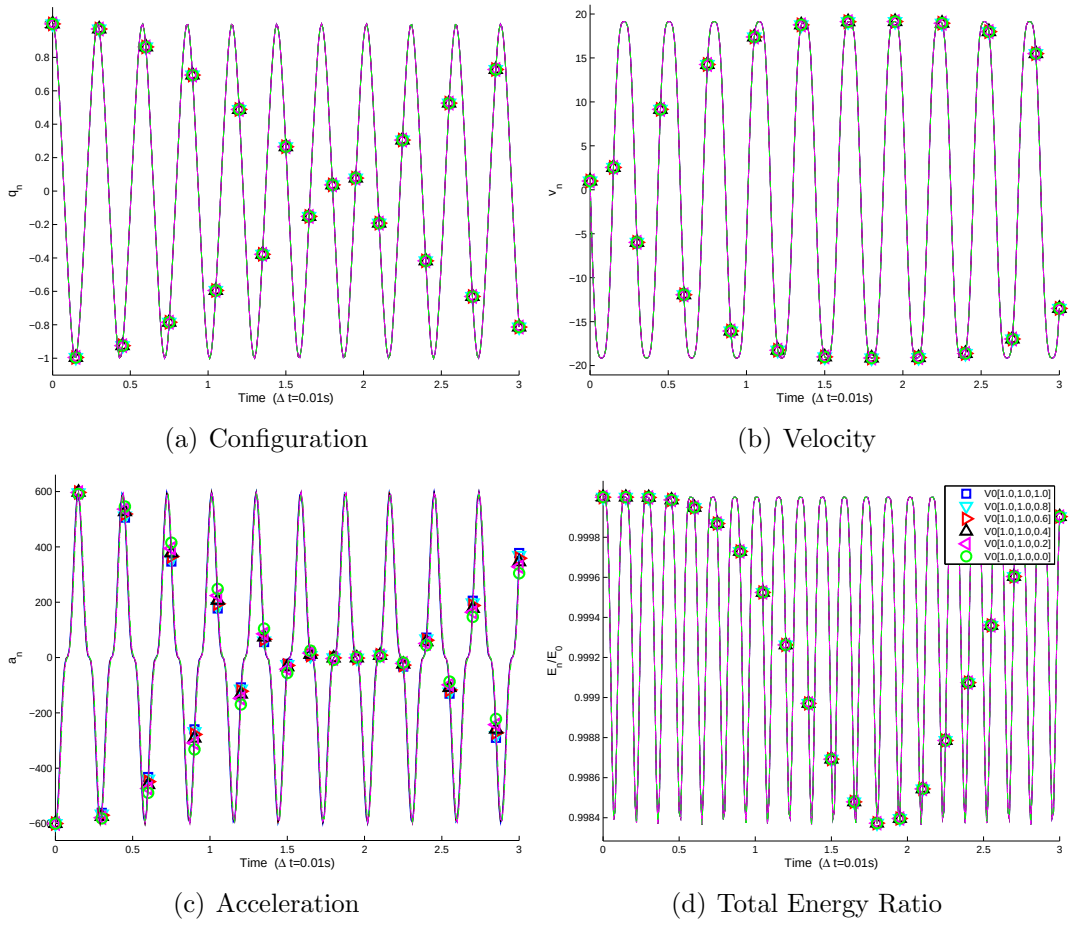
Figure A.21: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Hardening spring] [Implicit GSSSS family of algorithms (Option I) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

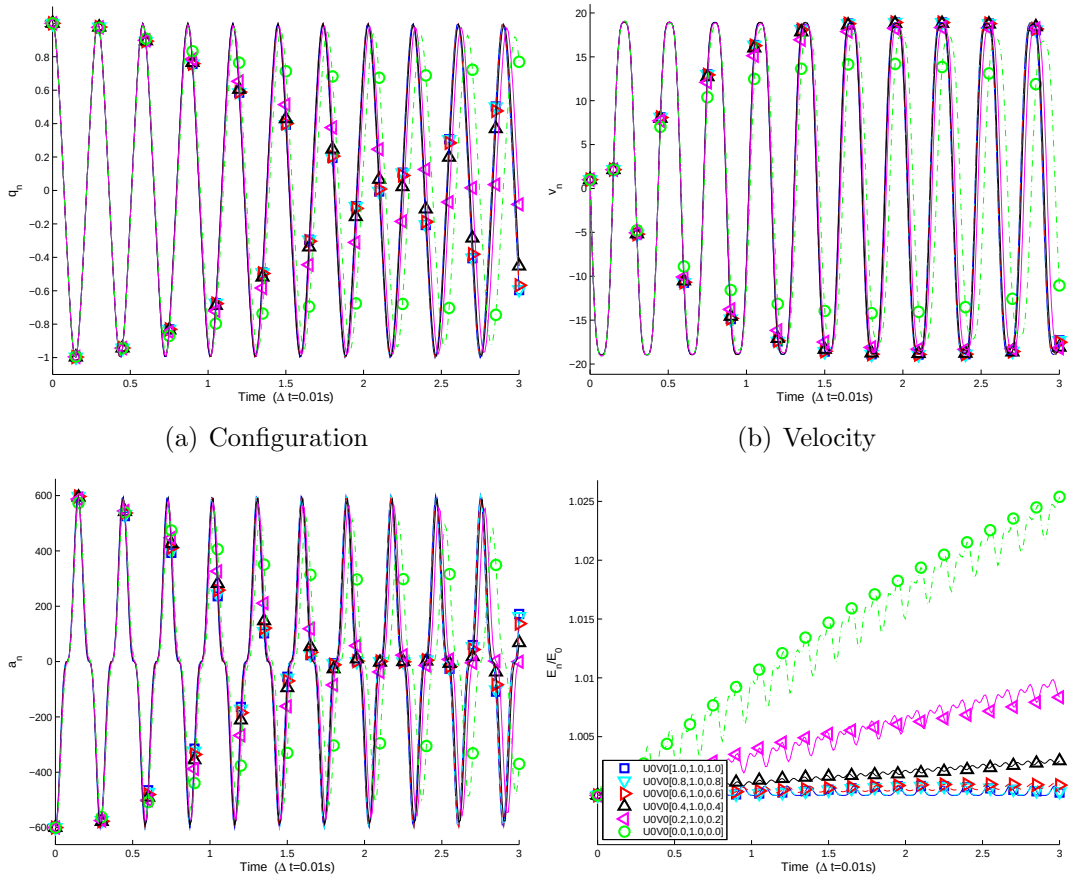
Figure A.22: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Hardening spring] [Implicit GSSSS family of algorithms (Option I) - U0V0(1.0,1.0,ρ^{s}_{∞})]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

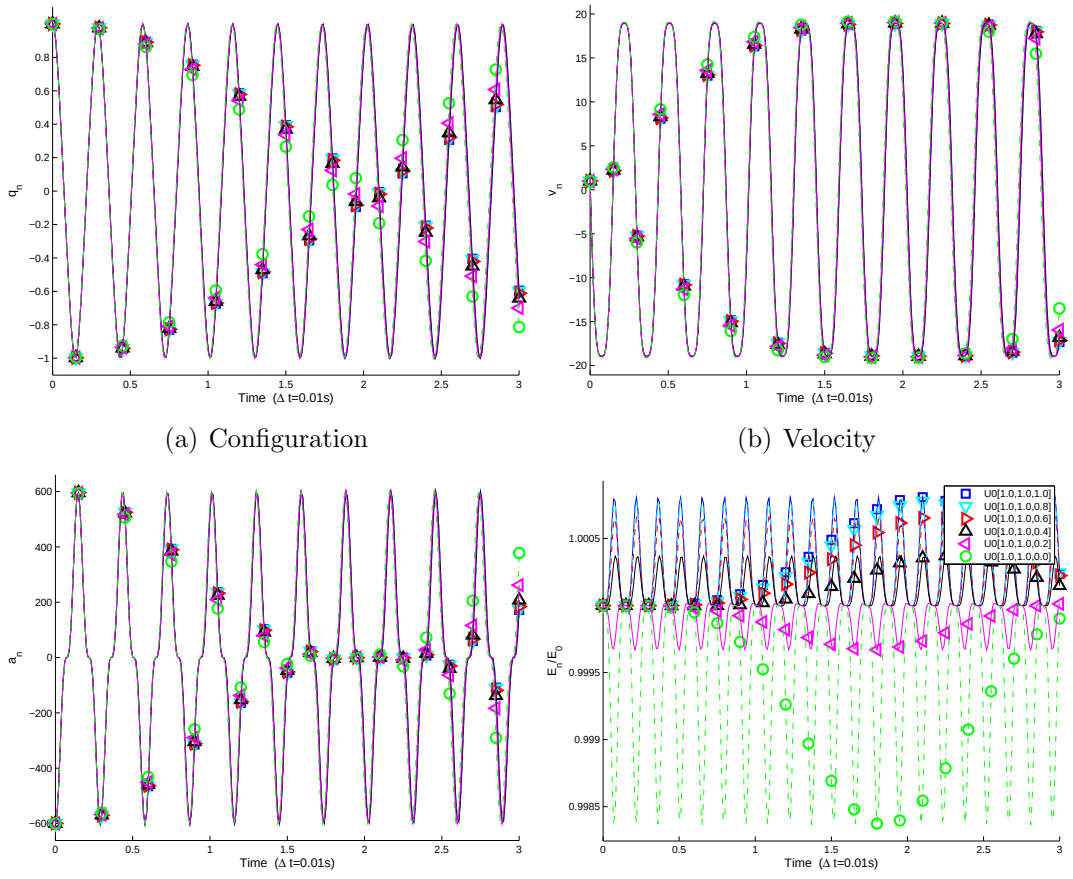
Figure A.23: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Hardening spring] [Implicit GSSSS family of algorithms (Option I) - V0U0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

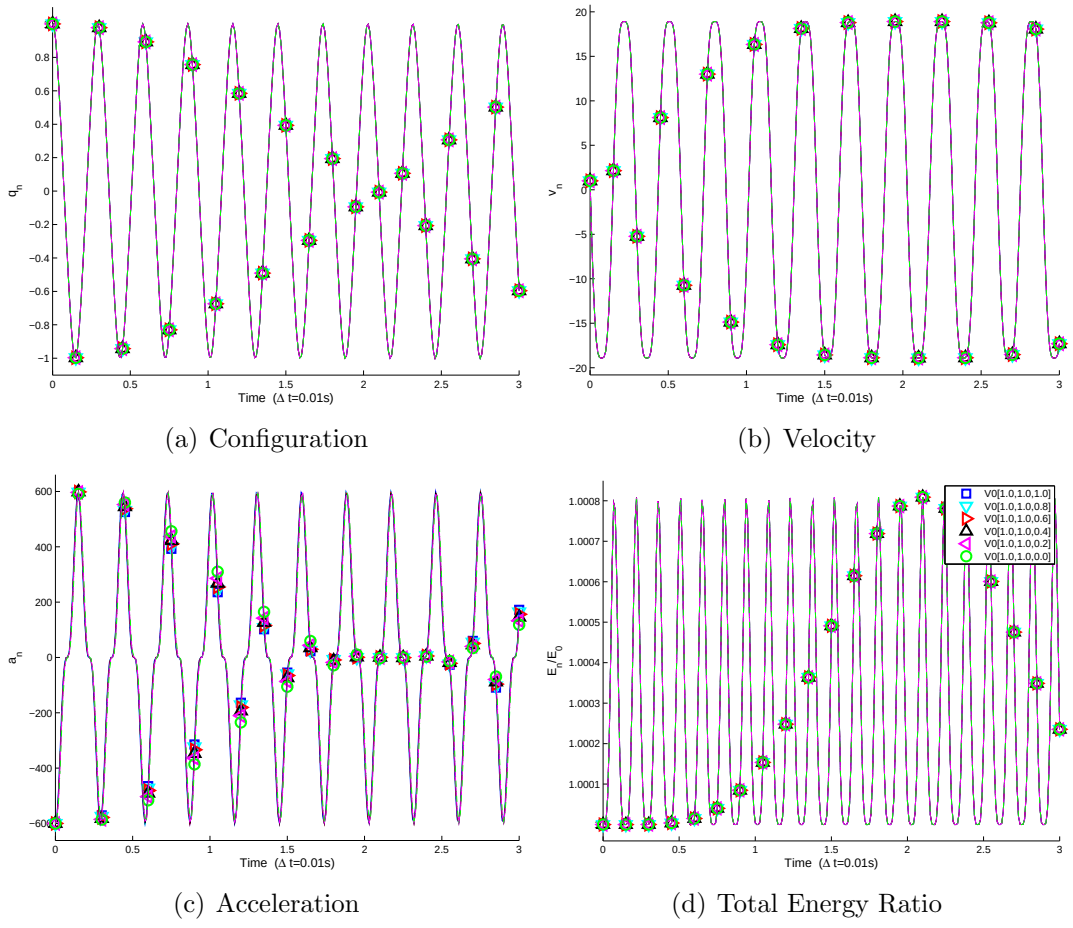
Figure A.24: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Hardening spring] [Implicit GSSSS family of algorithms (Option II) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

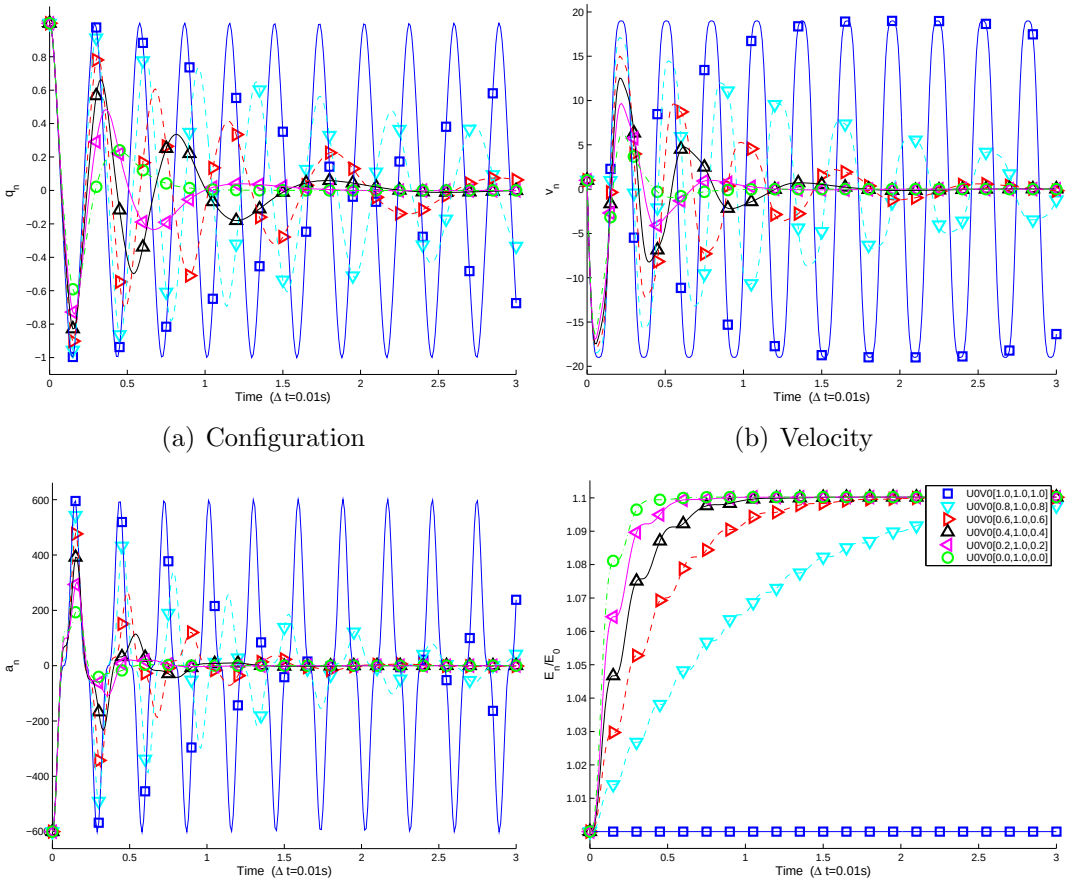
Figure A.25: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Hardening spring] [Implicit GSSSS family of algorithms (Option II) - U0V0(1.0,1.0,ρ^{s}_{∞})]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

Figure A.26: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Hardening spring] [Implicit GSSSS family of algorithms (Option II) - V0U0(1.0,1.0,ρ_∞^s)]



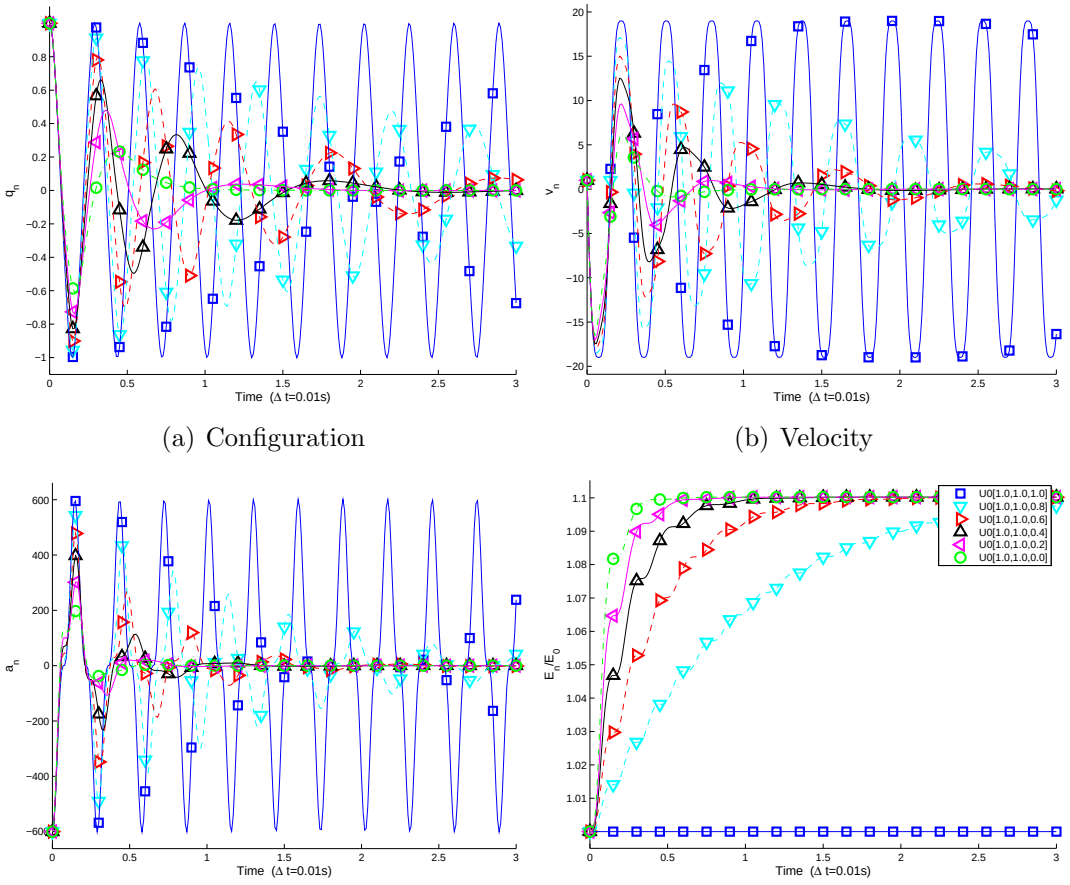
(a) Configuration

(b) Velocity

(c) Acceleration

(d) Total Energy Ratio

Figure A.27: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Hardening spring] [Implicit GSSSS family of algorithms (Option III) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



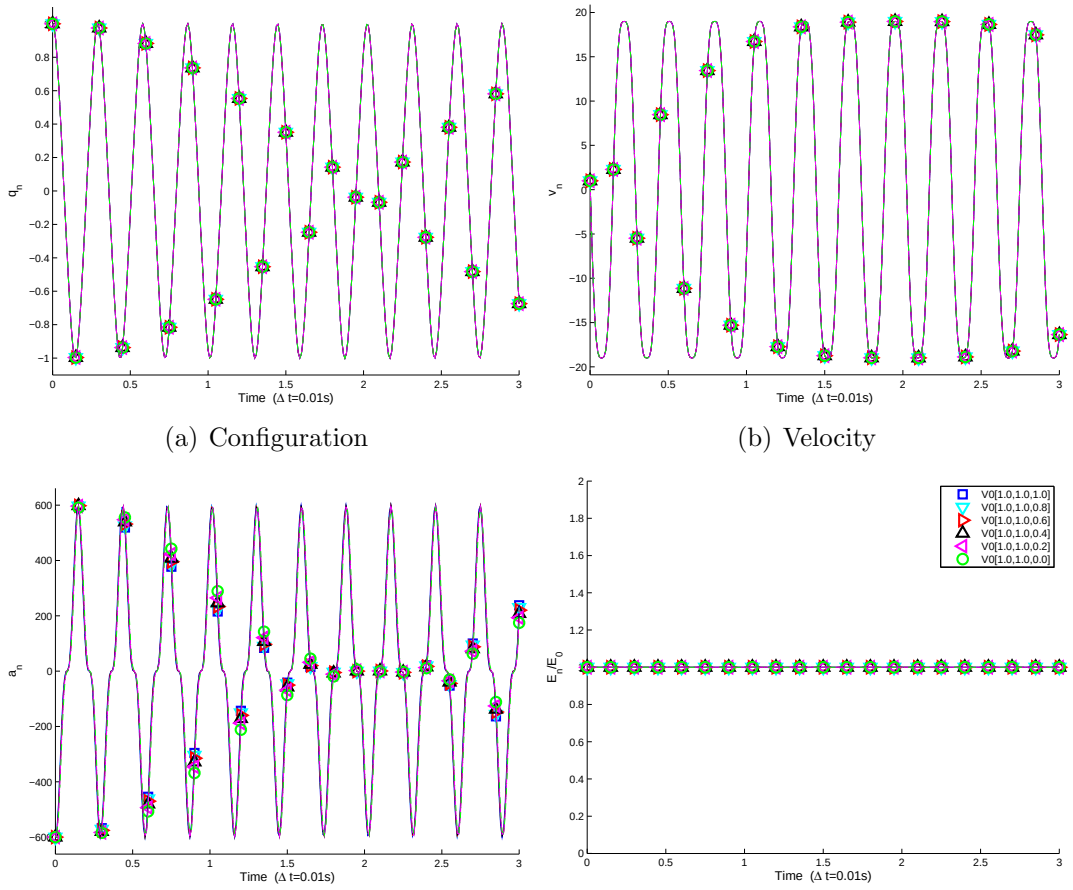
(a) Configuration

(b) Velocity

(c) Acceleration

(d) Total Energy Ratio

Figure A.28: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Hardening spring] [Implicit GSSSS family of algorithms (Option III) - U0V0(1.0,1.0,ρ_∞^s)]



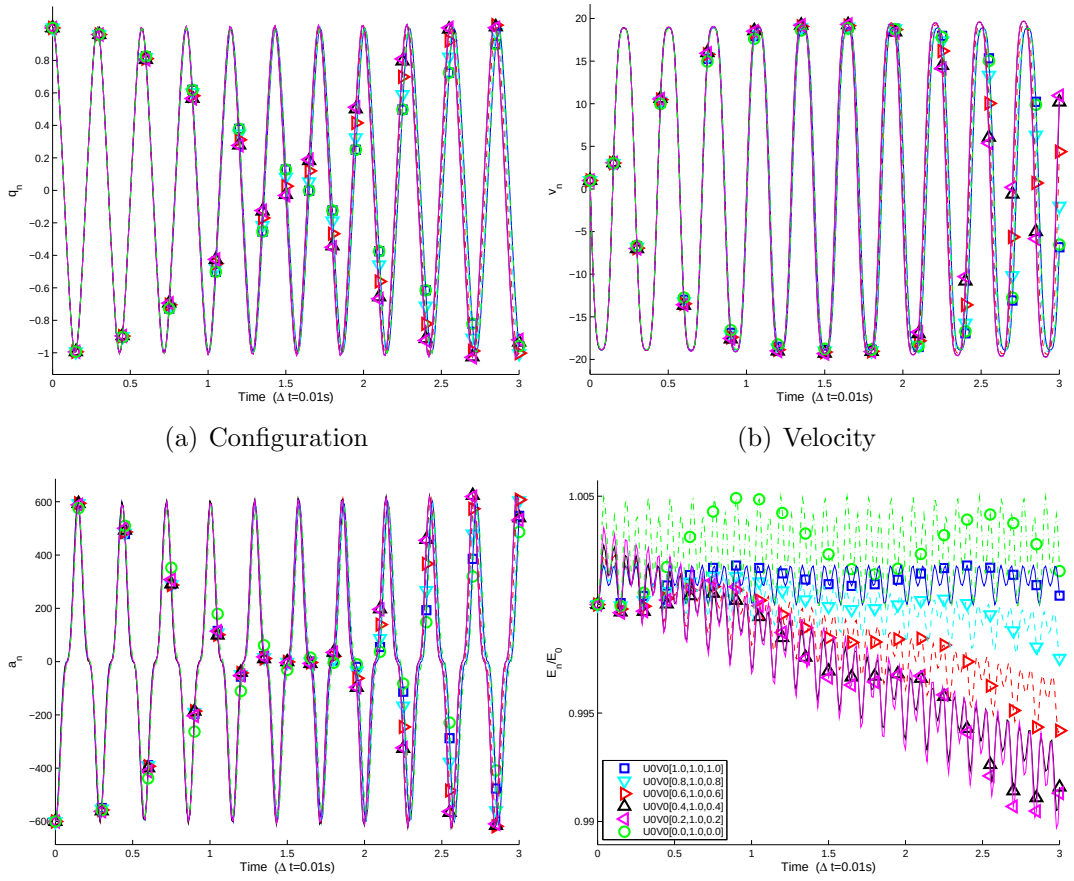
(a) Configuration

(b) Velocity

(c) Acceleration

(d) Total Energy Ratio

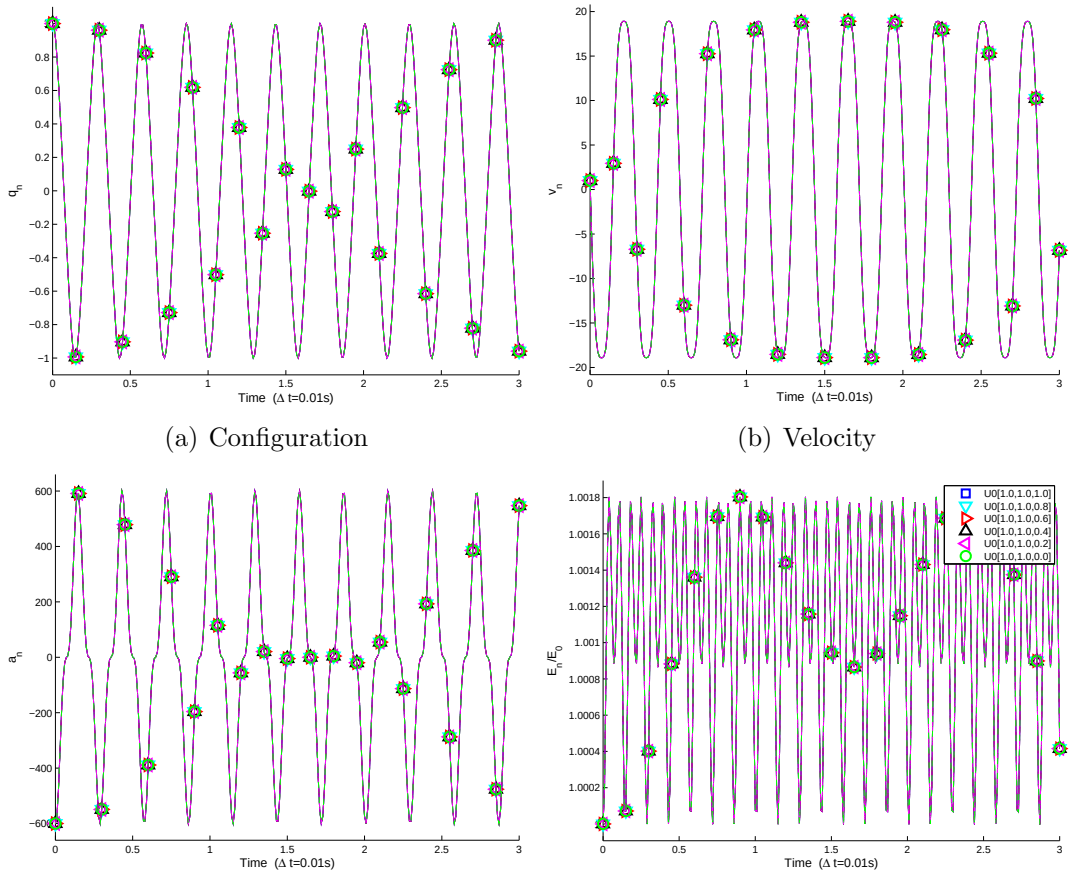
Figure A.29: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Hardening spring] [Implicit GSSSS family of algorithms (Option III) - V0U0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

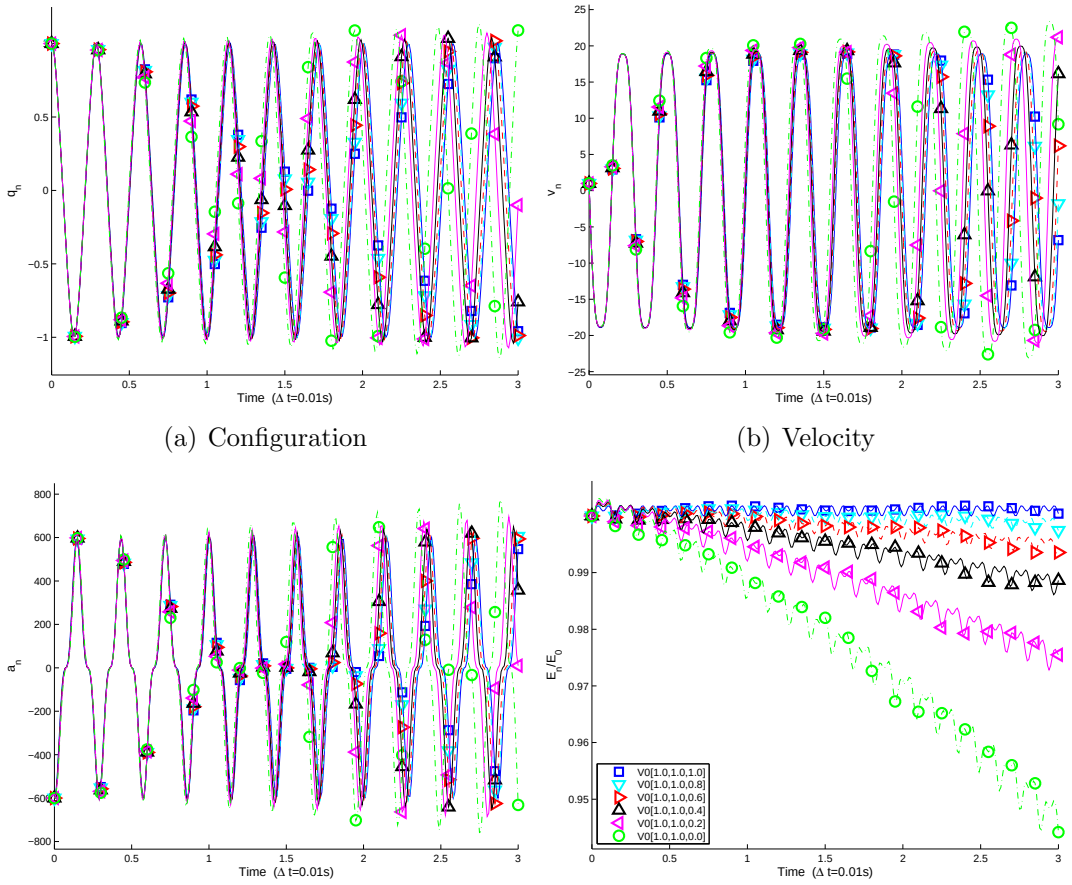
Figure A.30: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Hardening spring] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option I) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

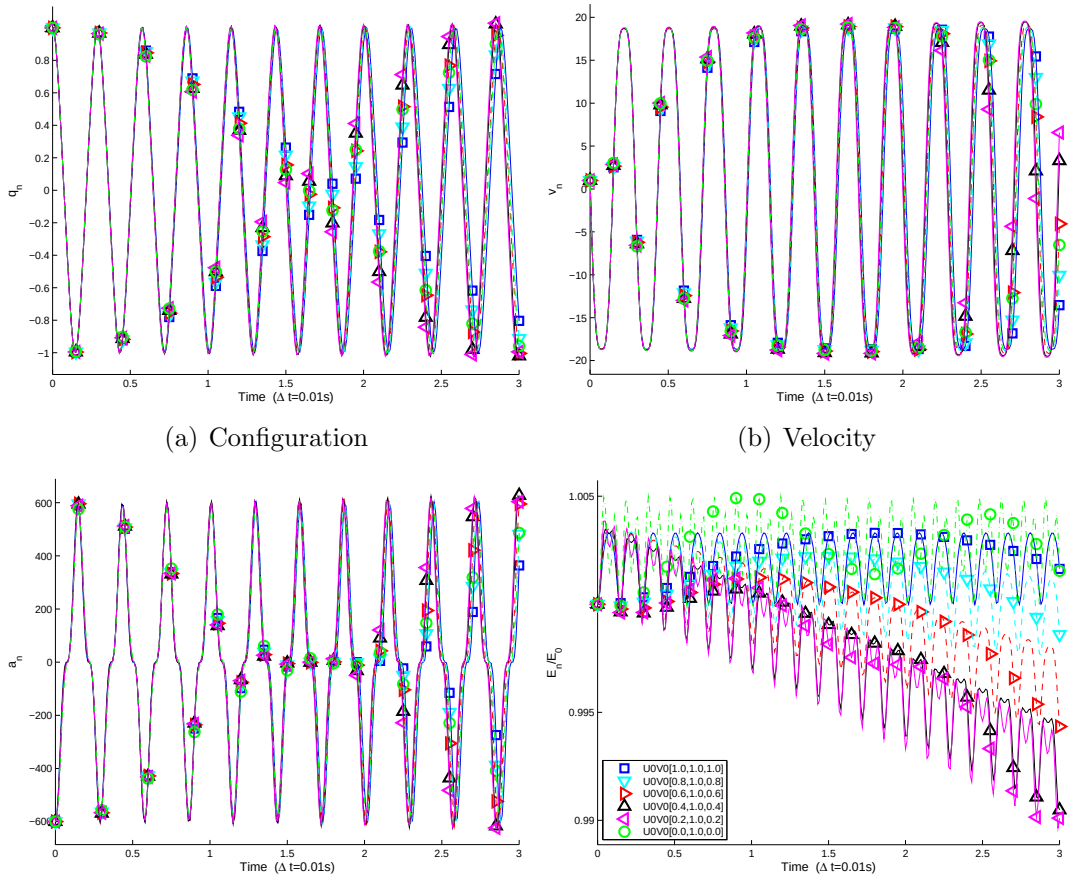
Figure A.31: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Hardening spring] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option I) - U0V0(1.0,1.0,ρ^s_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

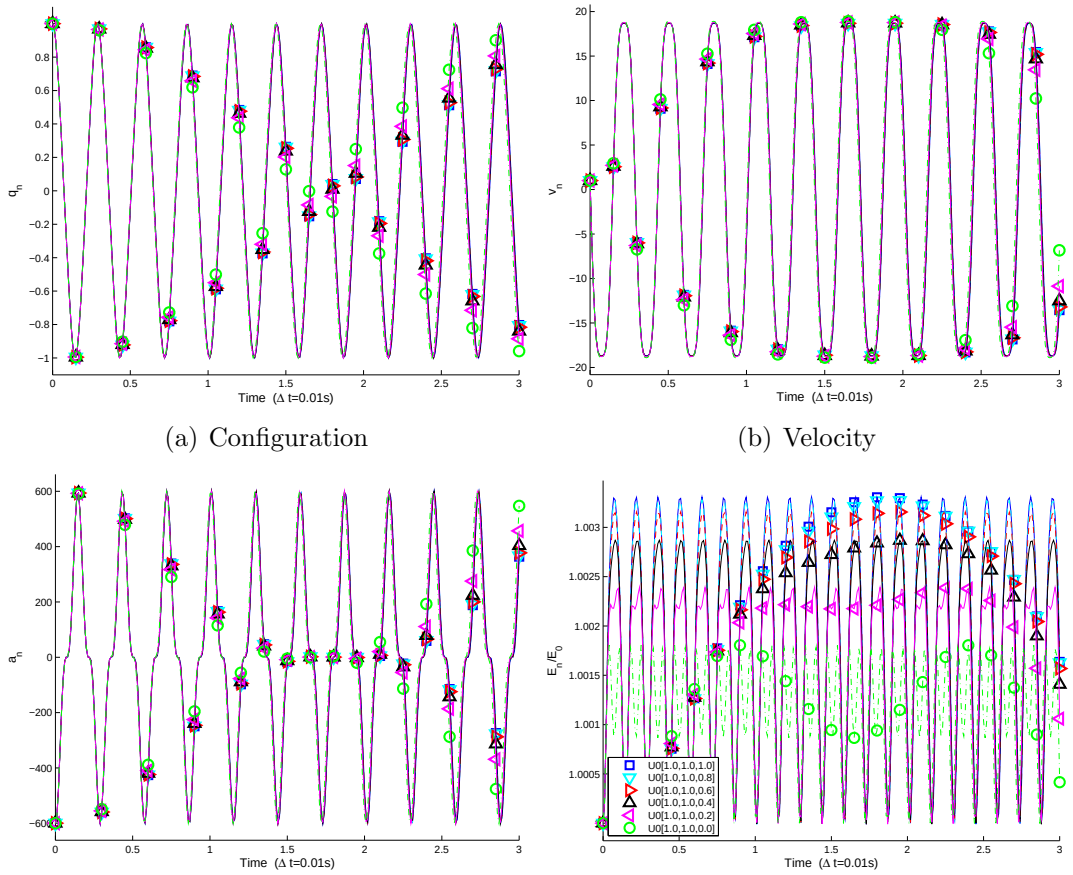
Figure A.32: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Hardening spring] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option I) - V0U0(1.0,1.0,ρ^s_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

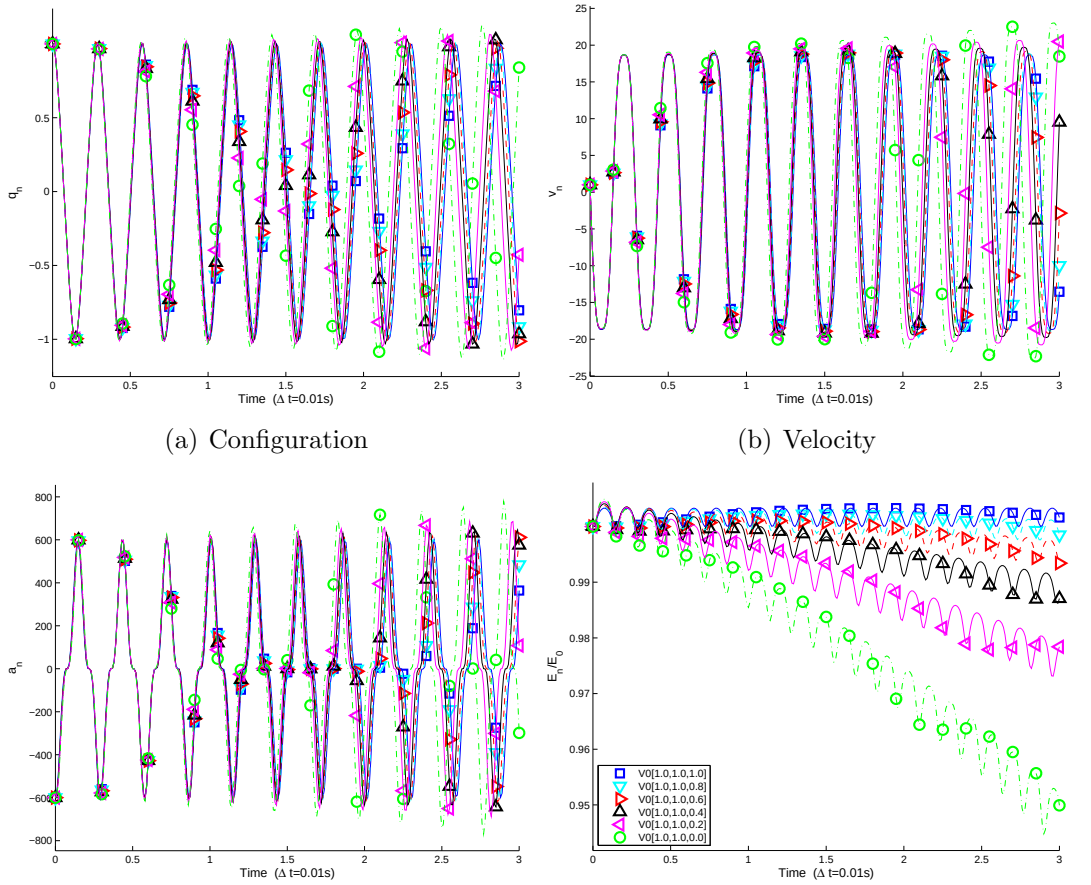
Figure A.33: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Hardening spring] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option II) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

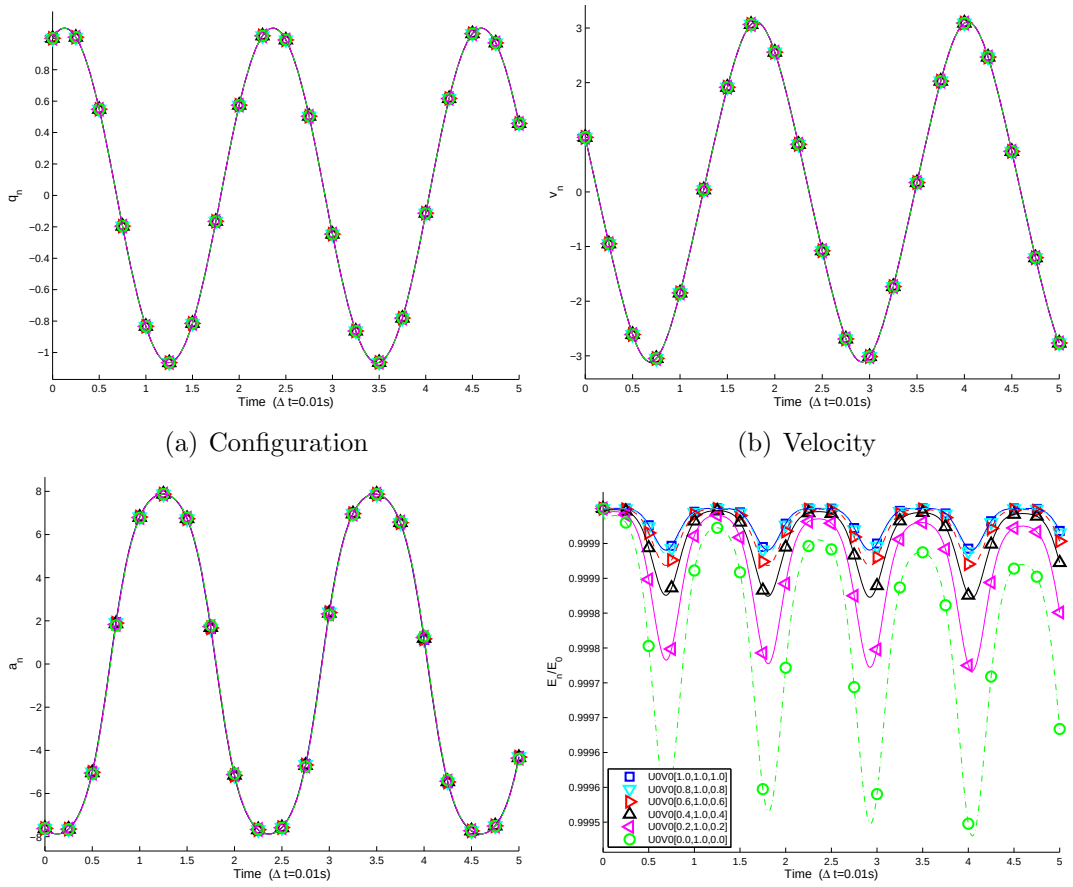
Figure A.34: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Hardening spring] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option II) - U0V0(1.0,1.0,ρ_∞^{s})]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

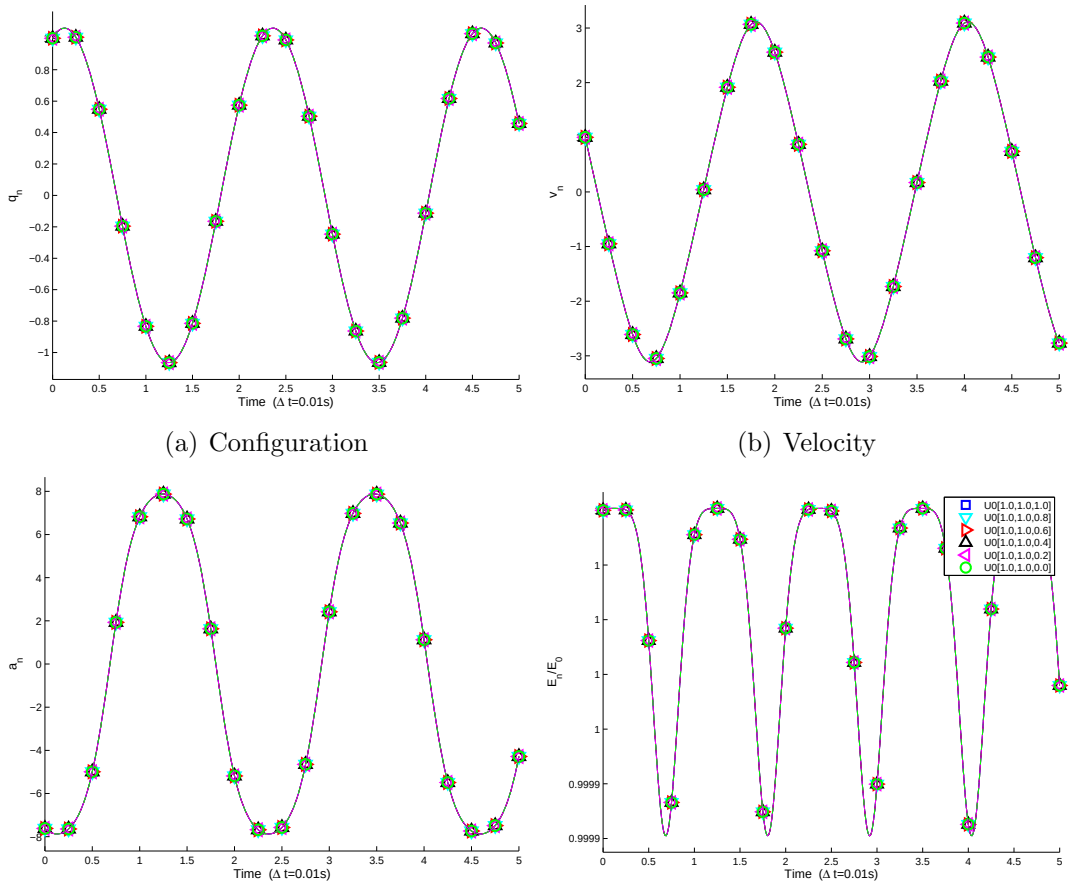
Figure A.35: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Hardening spring] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option II) - V0U0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

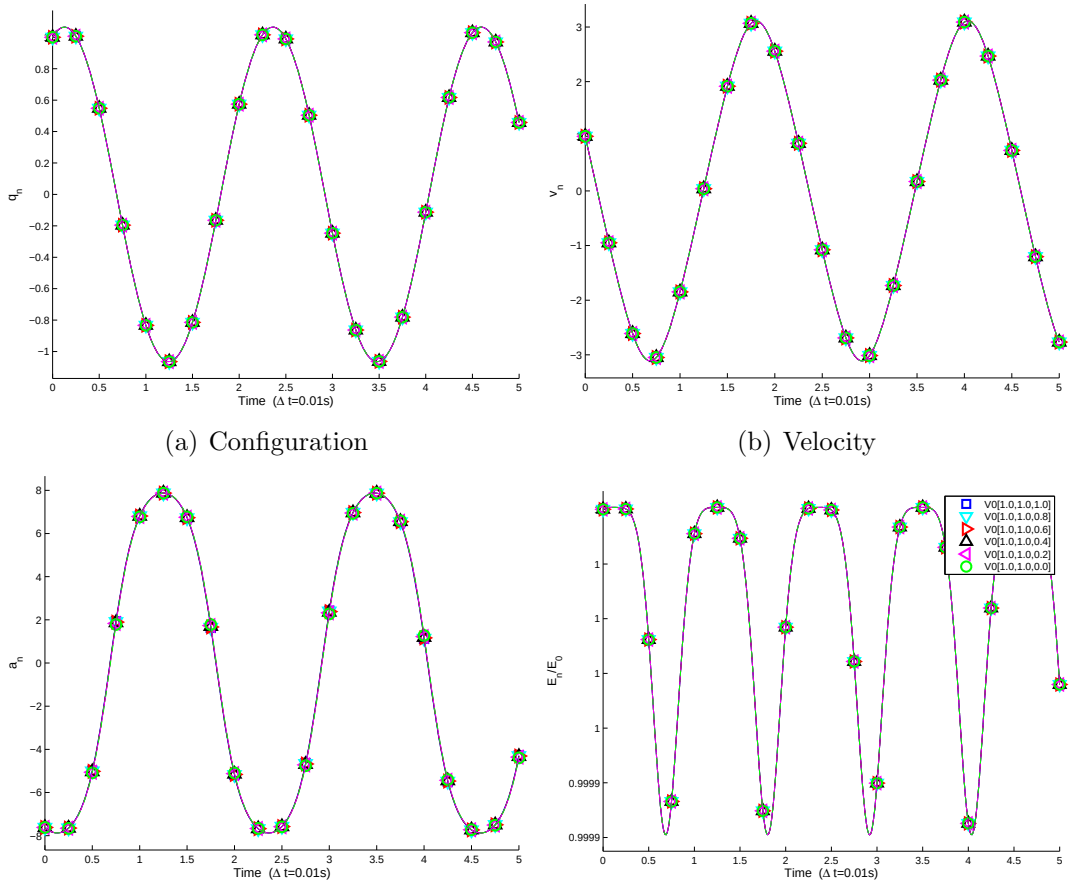
Figure A.36: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Biliner softening spring] [Implicit GSSSS family of algorithms (Option I) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

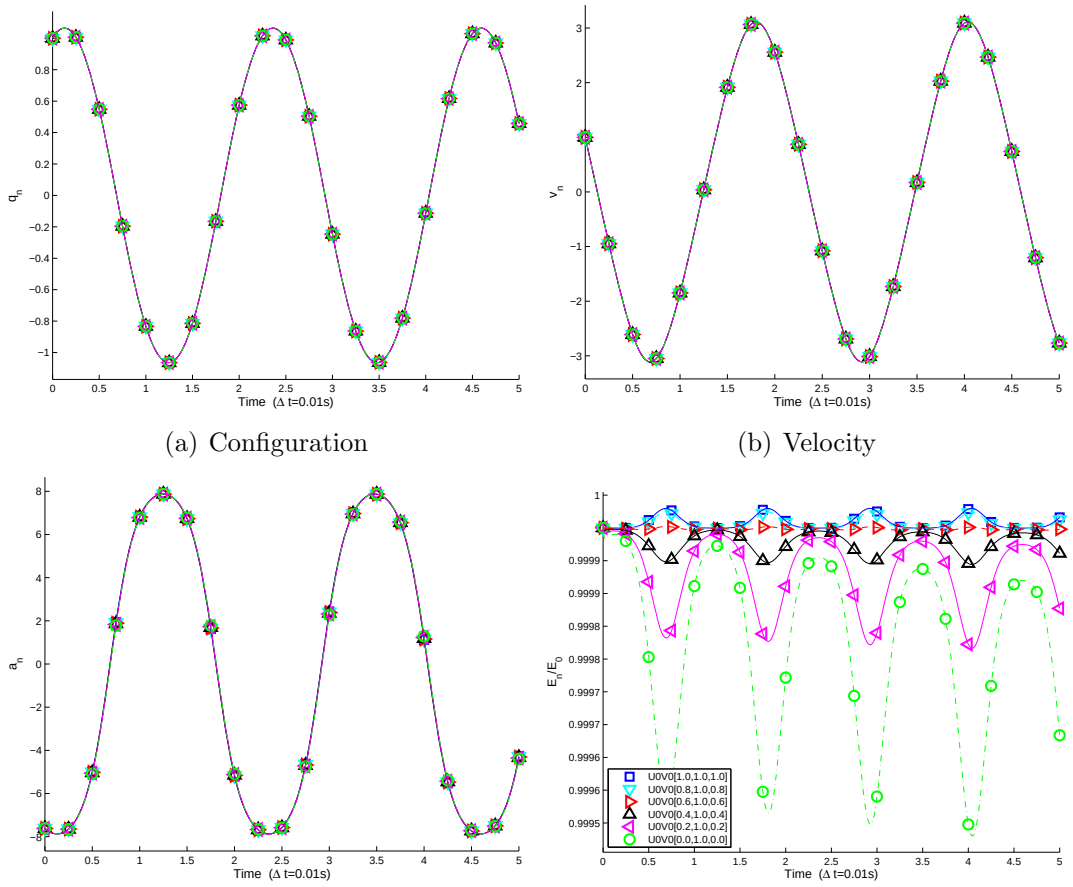
Figure A.37: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Biliner softening spring] [Implicit GSSSS family of algorithms (Option I) - U0V0(1.0,1.0,ρ^{s}_{∞})]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

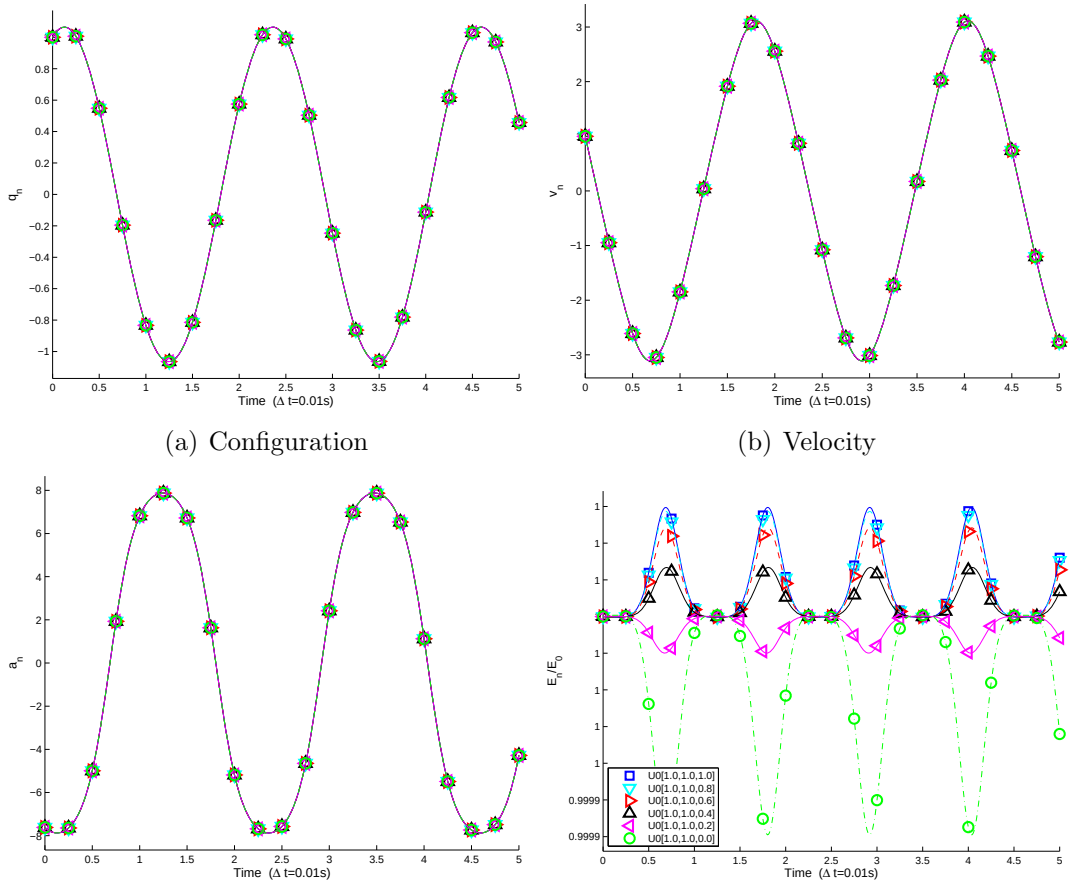
Figure A.38: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Biliner softening spring] [Implicit GSSSS family of algorithms (Option I) - V0U0(1.0,1.0,ρ^{s}_{∞})]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

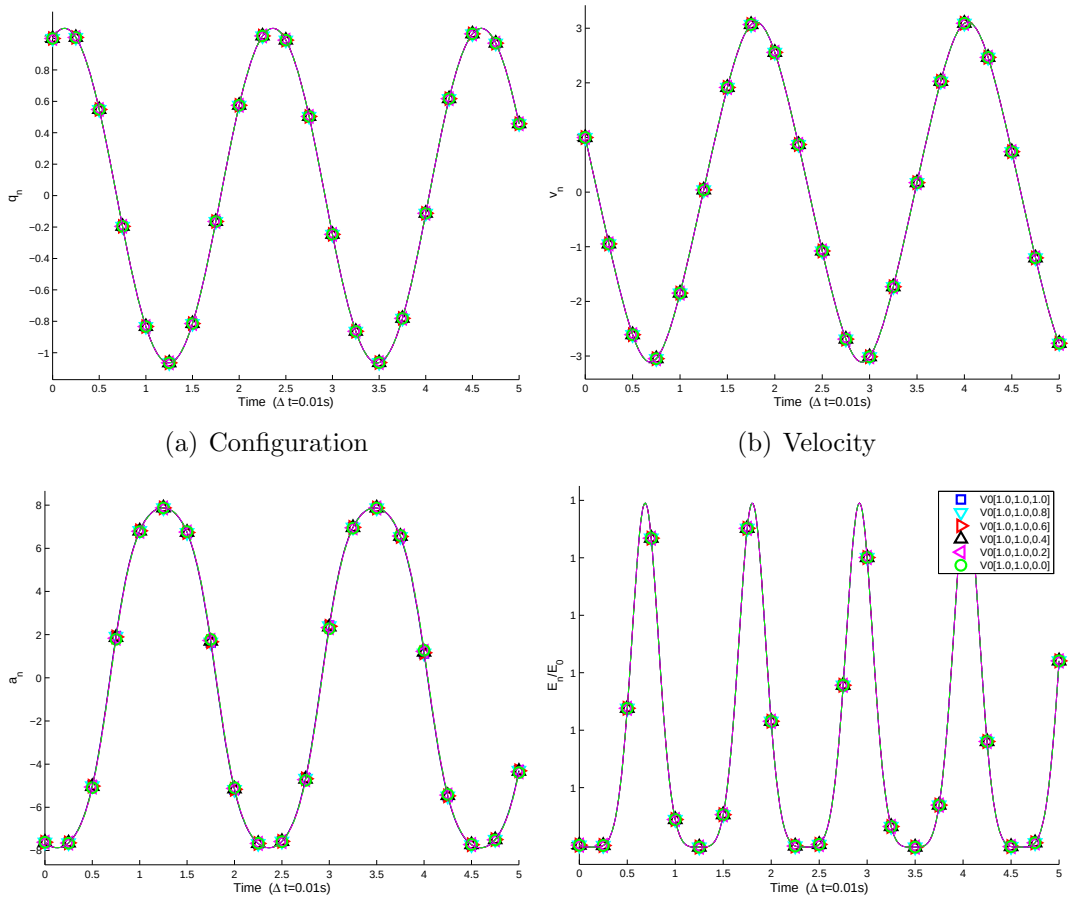
Figure A.39: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Biliner softening spring] [Implicit GSSSS family of algorithms (Option II) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

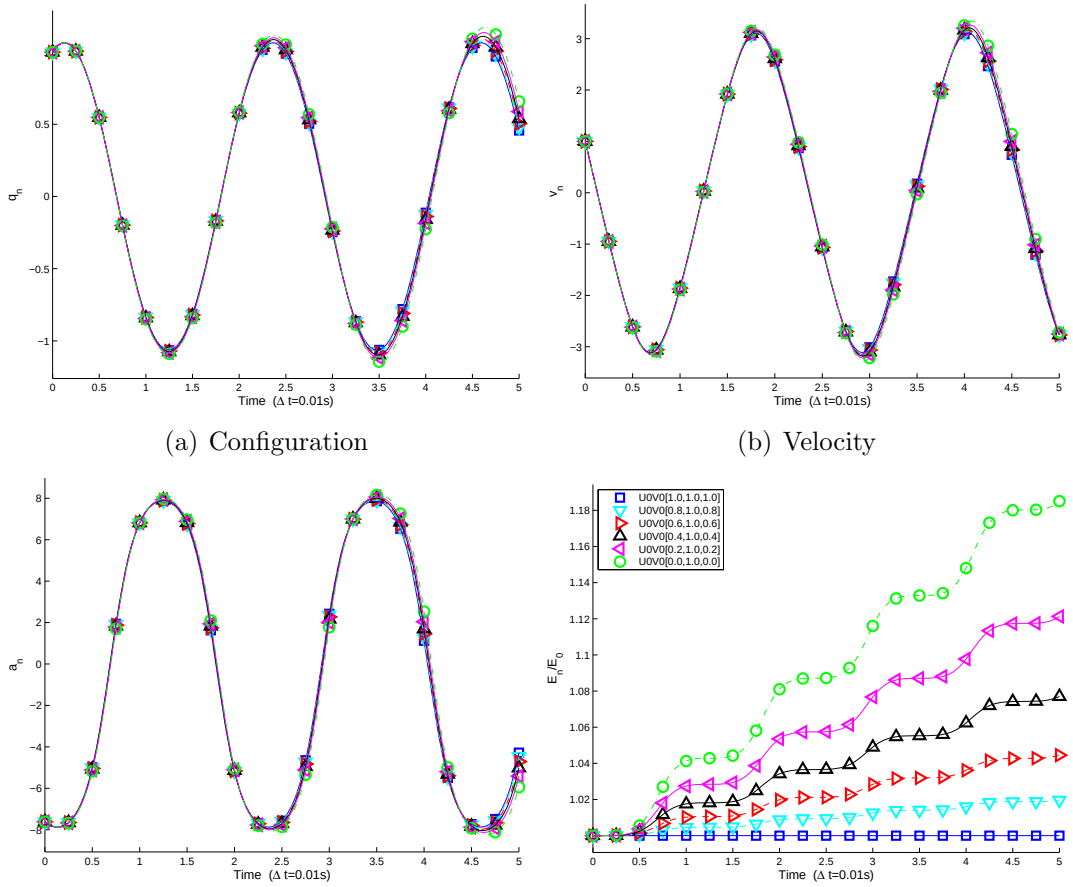
Figure A.40: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Biliner softening spring] [Implicit GSSSS family of algorithms (Option II) - U0V0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

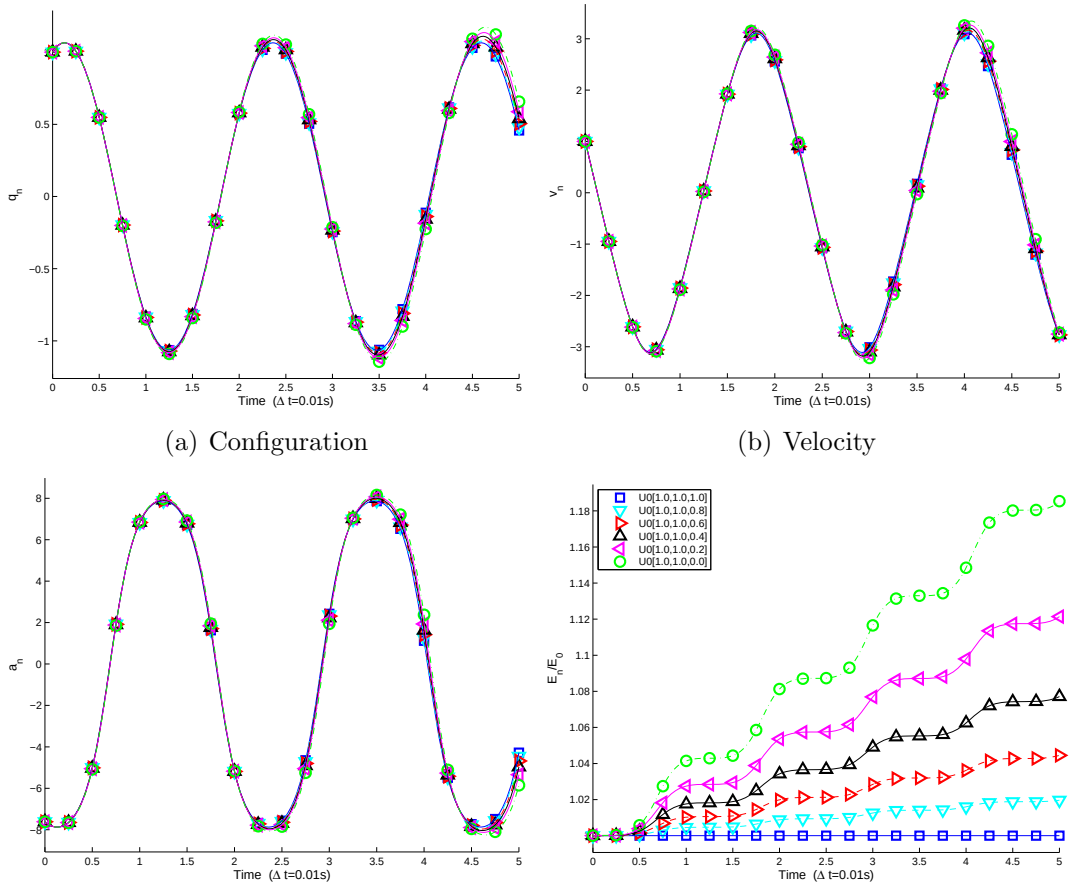
Figure A.41: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Biliner softening spring] [Implicit GSSSS family of algorithms (Option II) - V0U0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

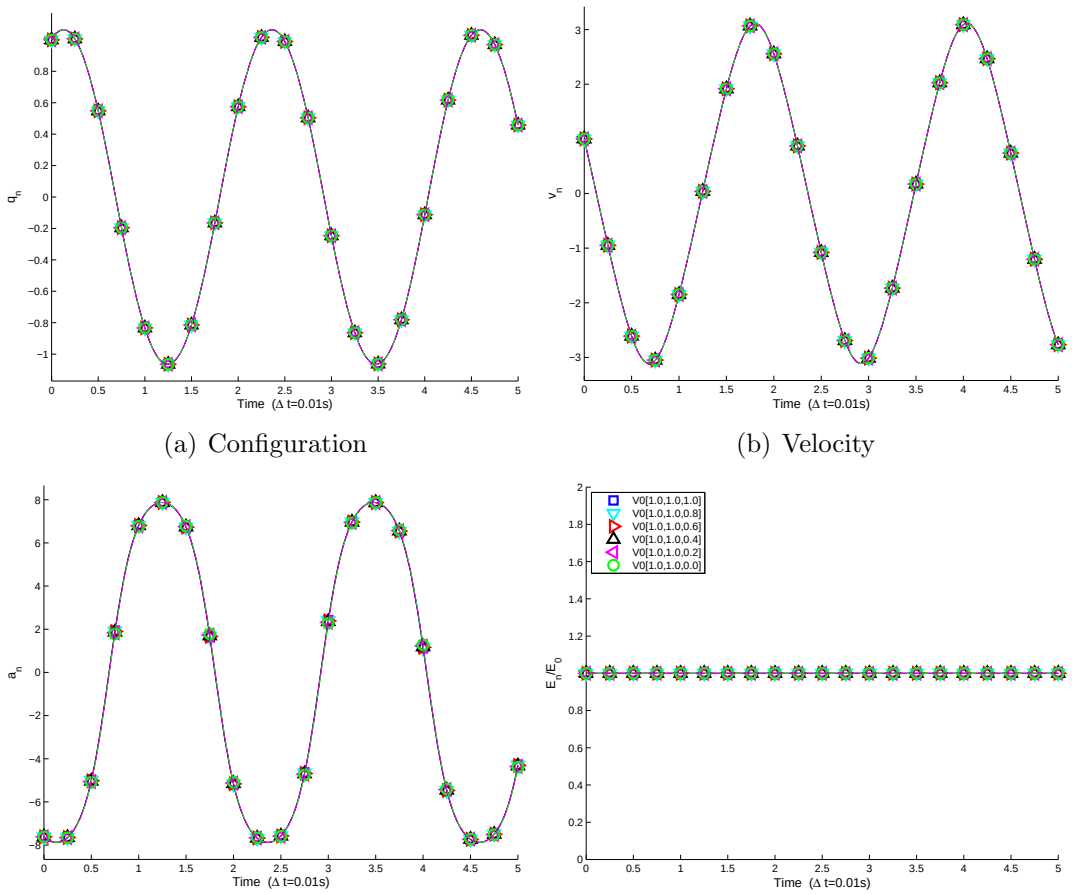
Figure A.42: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Biliner softening spring] [Implicit GSSSS family of algorithms (Option III) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

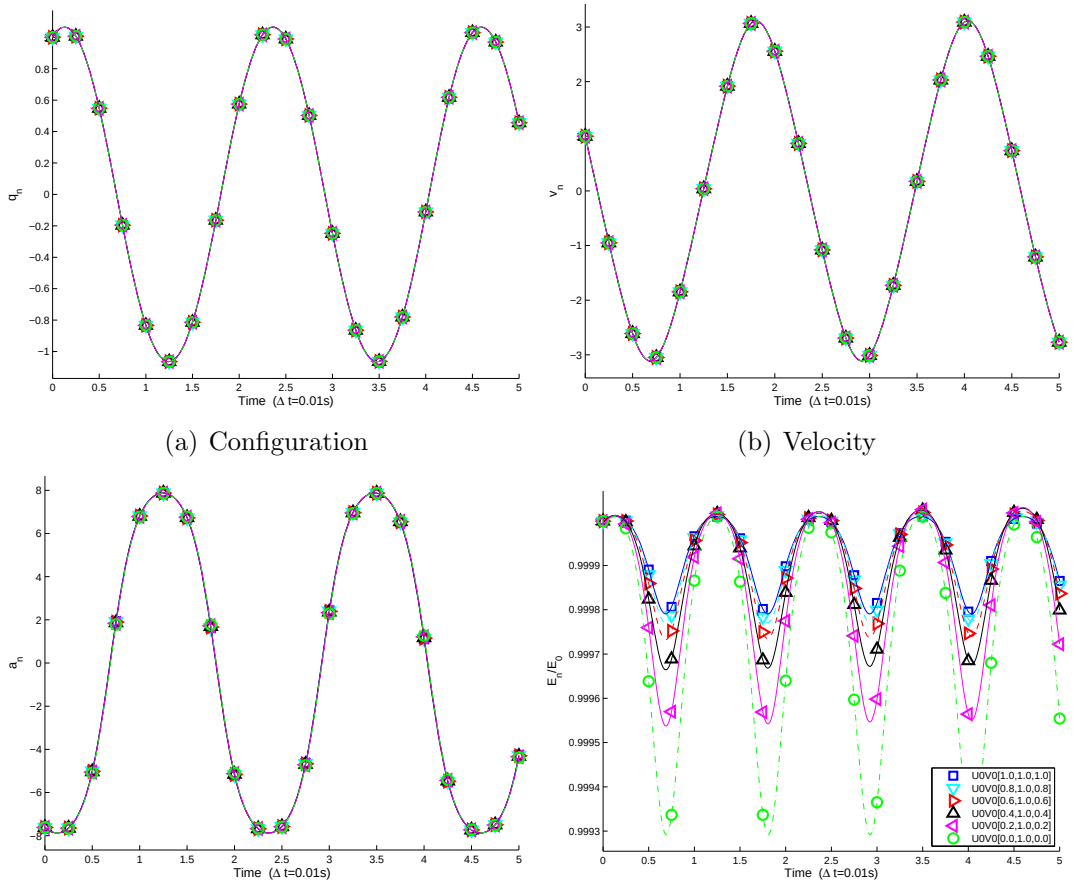
Figure A.43: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Biliner softening spring] [Implicit GSSSS family of algorithms (Option III) - U0V0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

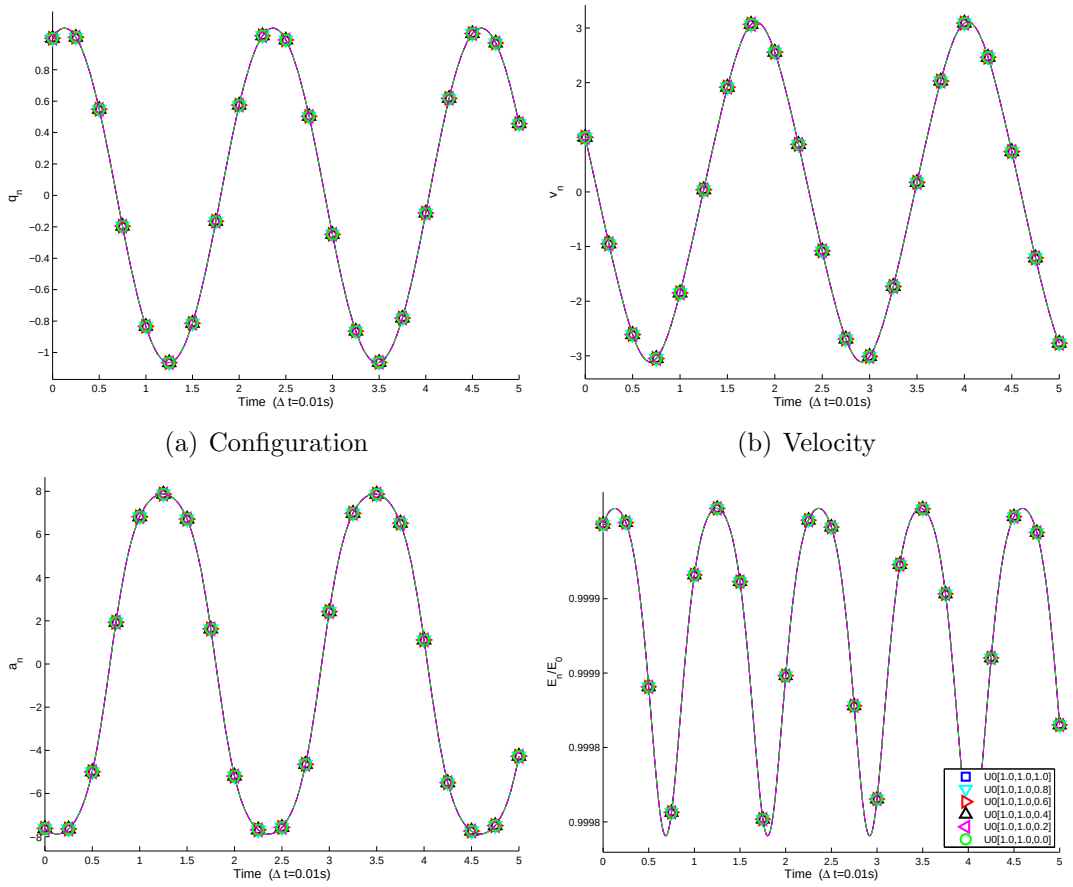
Figure A.44: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Biliner softening spring] [Implicit GSSSS family of algorithms (Option III) - V0U0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

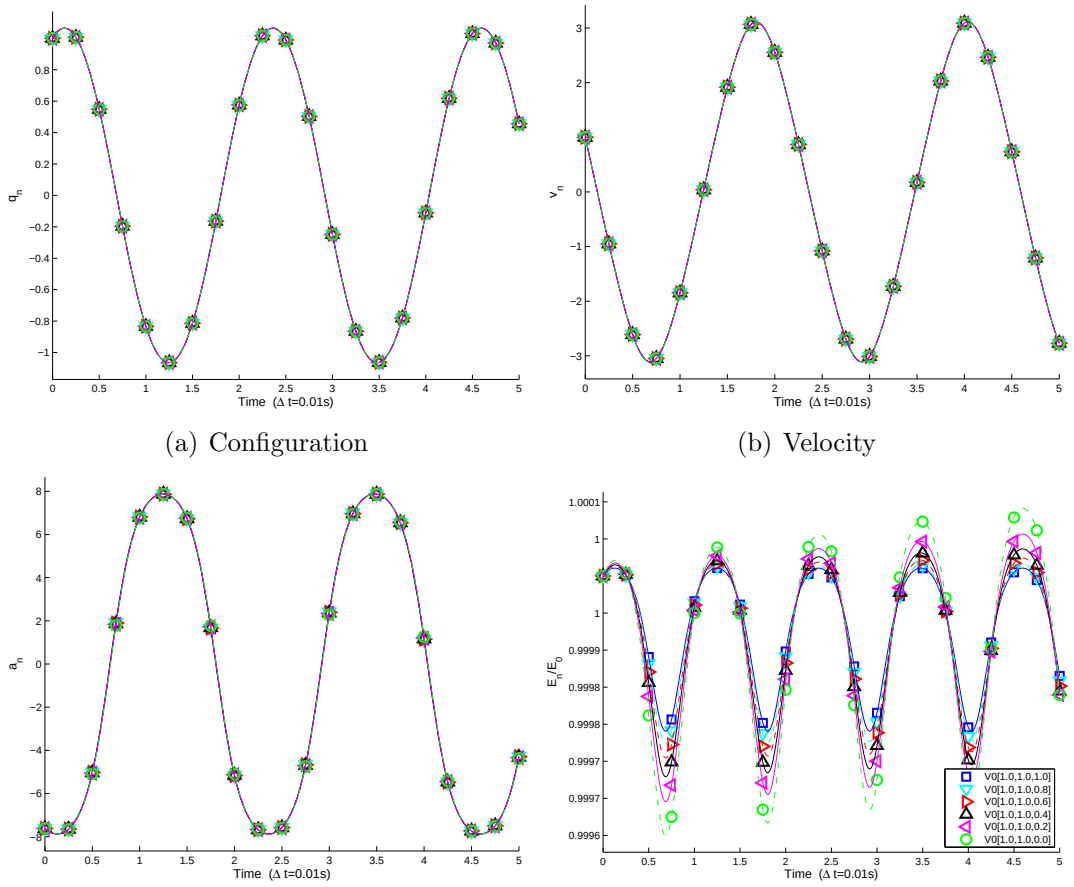
Figure A.45: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Biliner softening spring] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option I) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

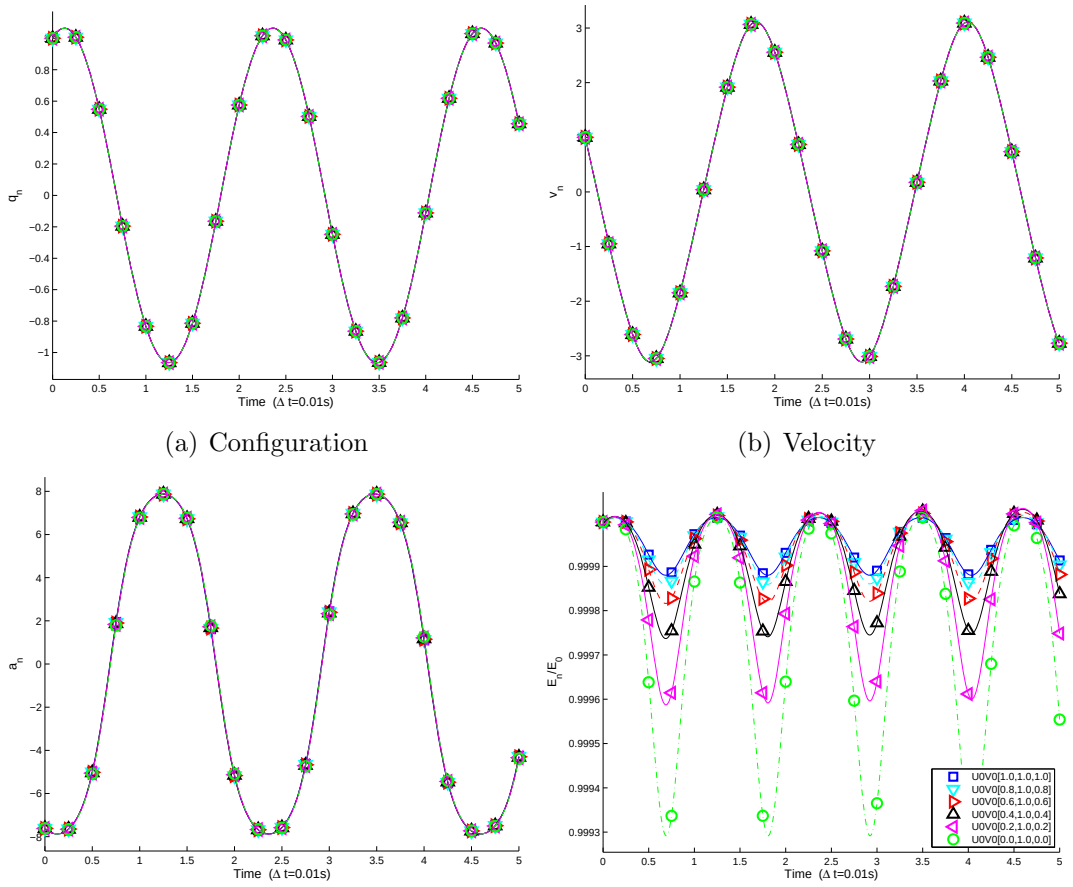
Figure A.46: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Biliner softening spring] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option I) - U0V0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

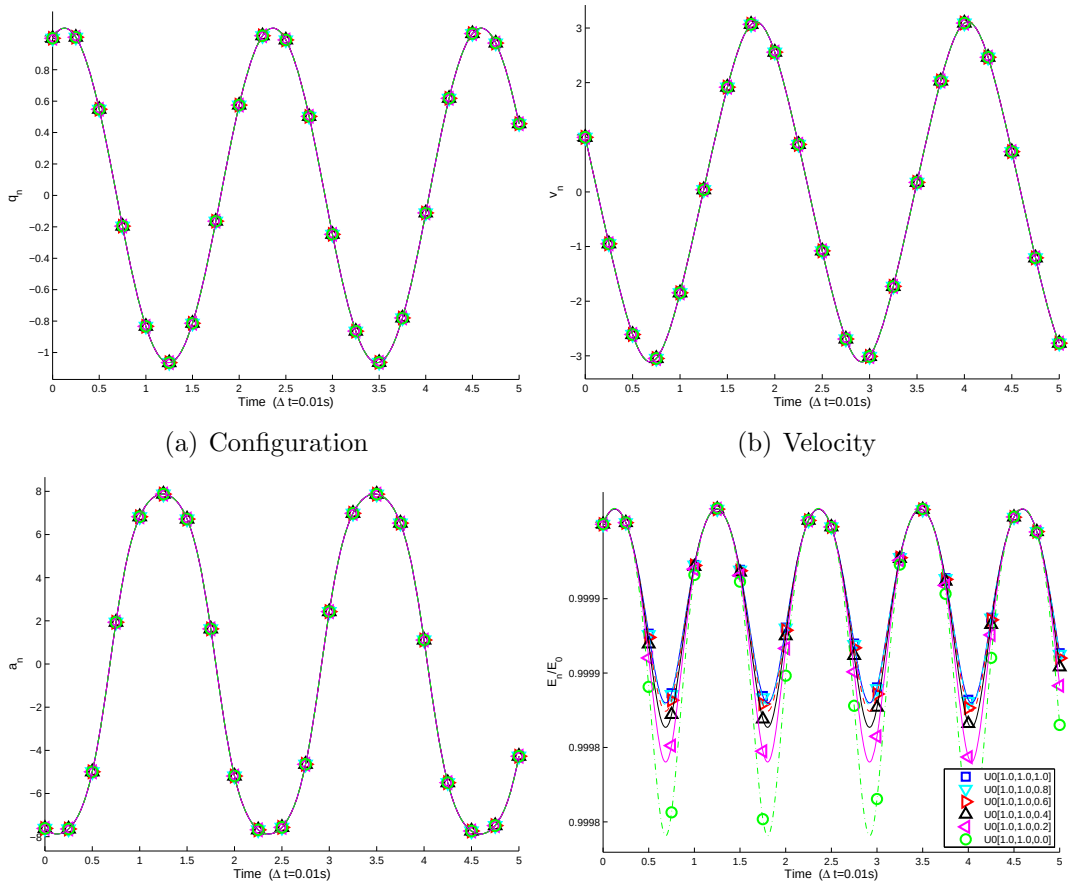
Figure A.47: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Biliner softening spring] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option I) - V0U0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

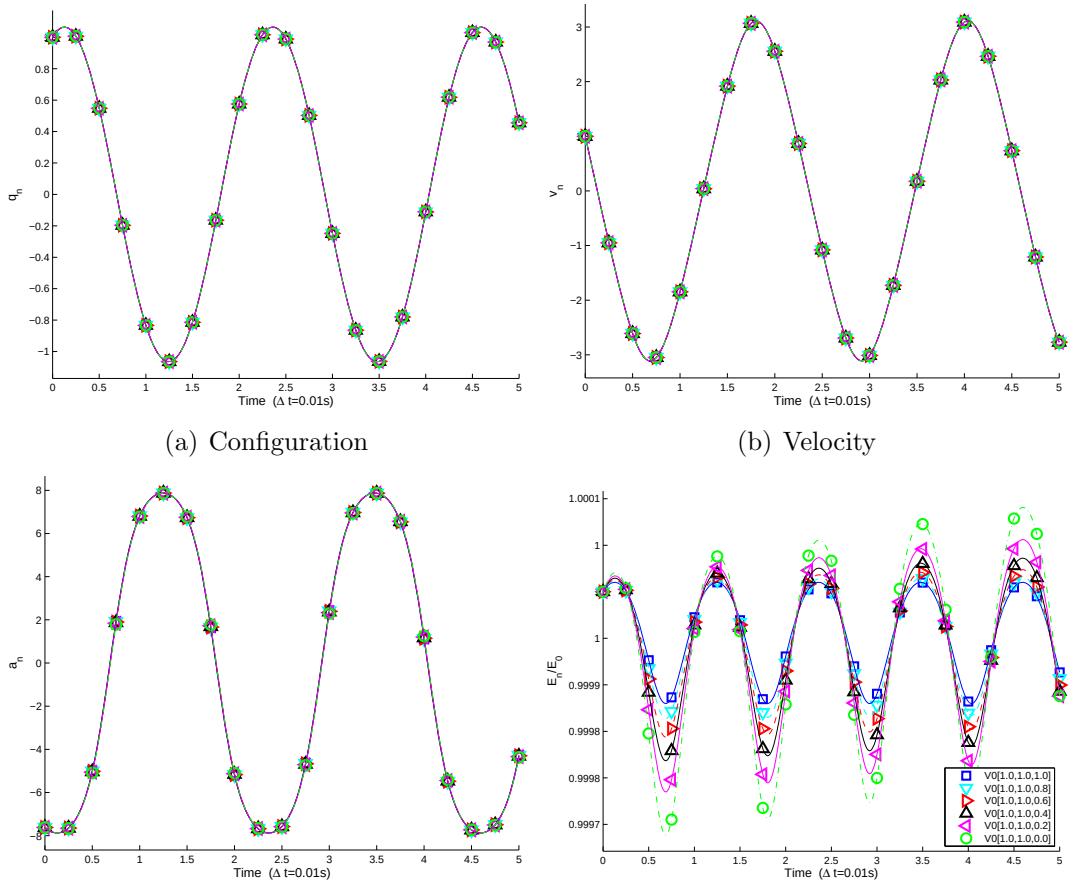
Figure A.48: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Biliner softening spring] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option II) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

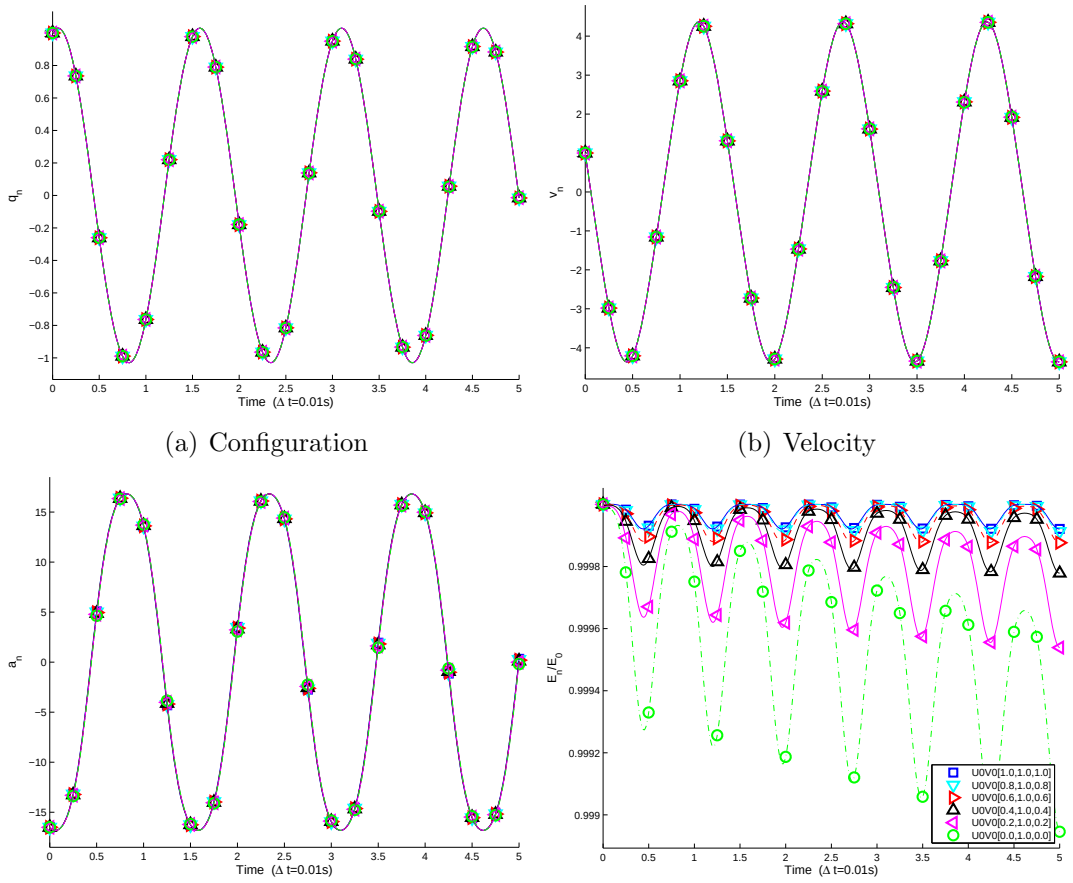
Figure A.49: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Biliner softening spring] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option II) - U0V0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

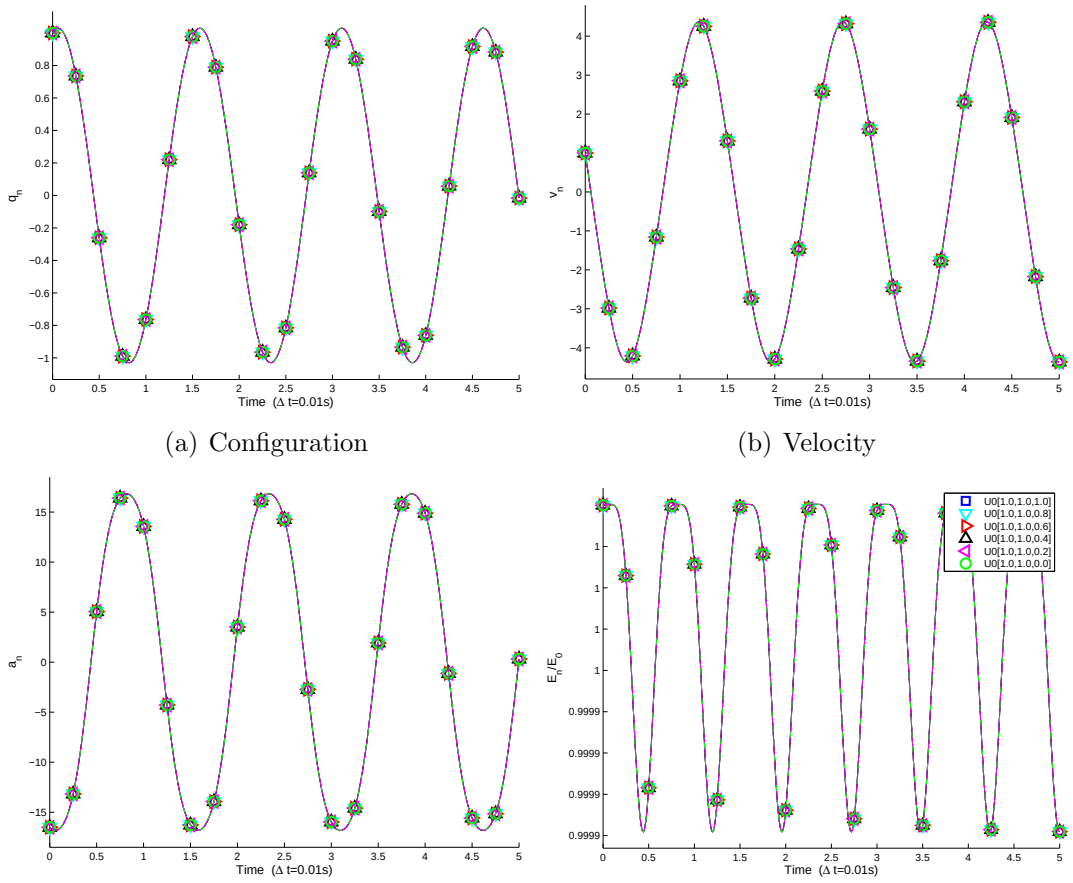
Figure A.50: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Biliner softening spring] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option II) - V0U0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

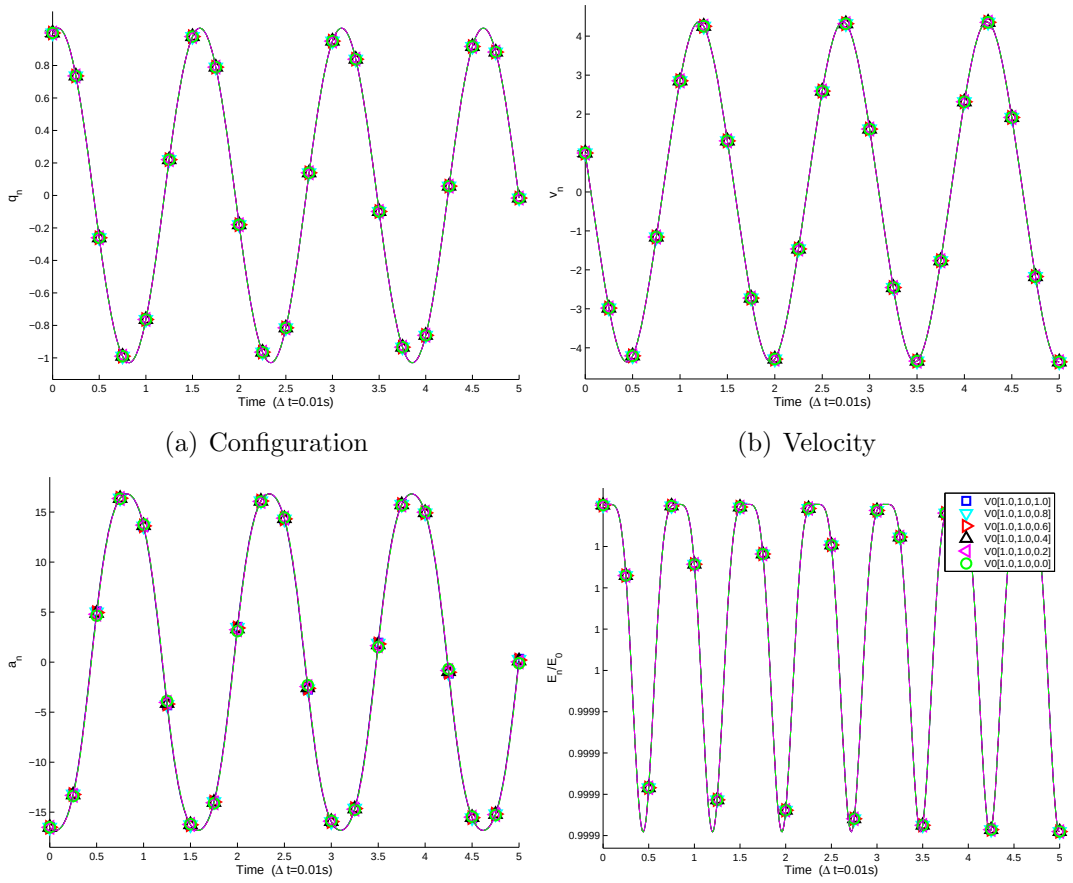
Figure A.51: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Simple pendulum] [Implicit GSSSS family of algorithms (Option I) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

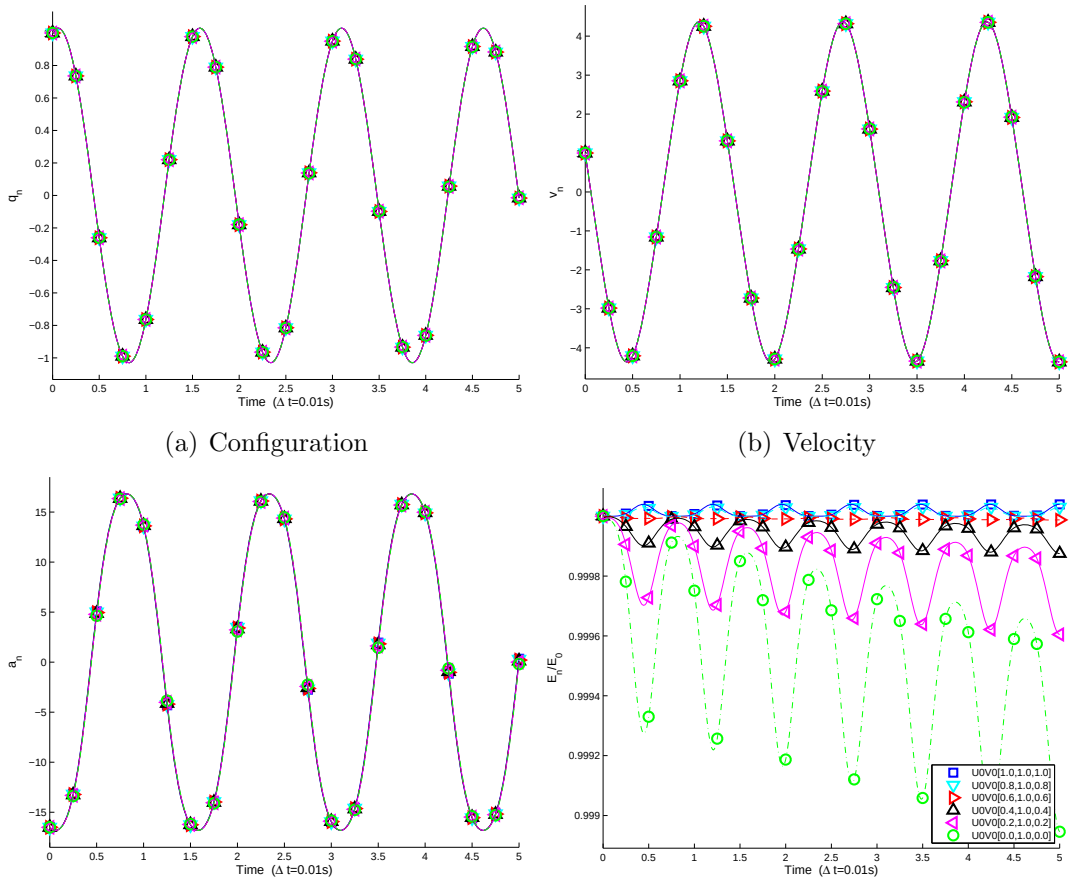
Figure A.52: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Simple pendulum] [Implicit GSSSS family of algorithms (Option I) - U0V0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

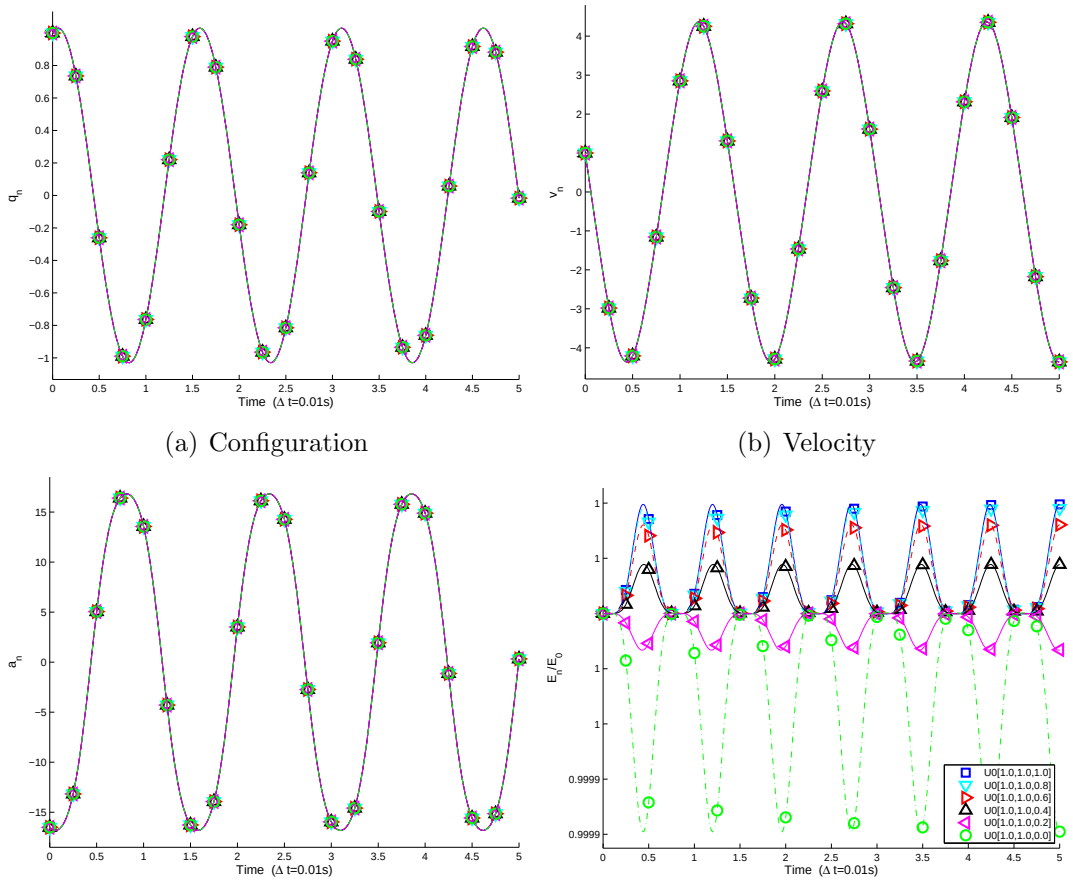
Figure A.53: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Simple pendulum] [Implicit GSSSS family of algorithms (Option I) - V0U0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

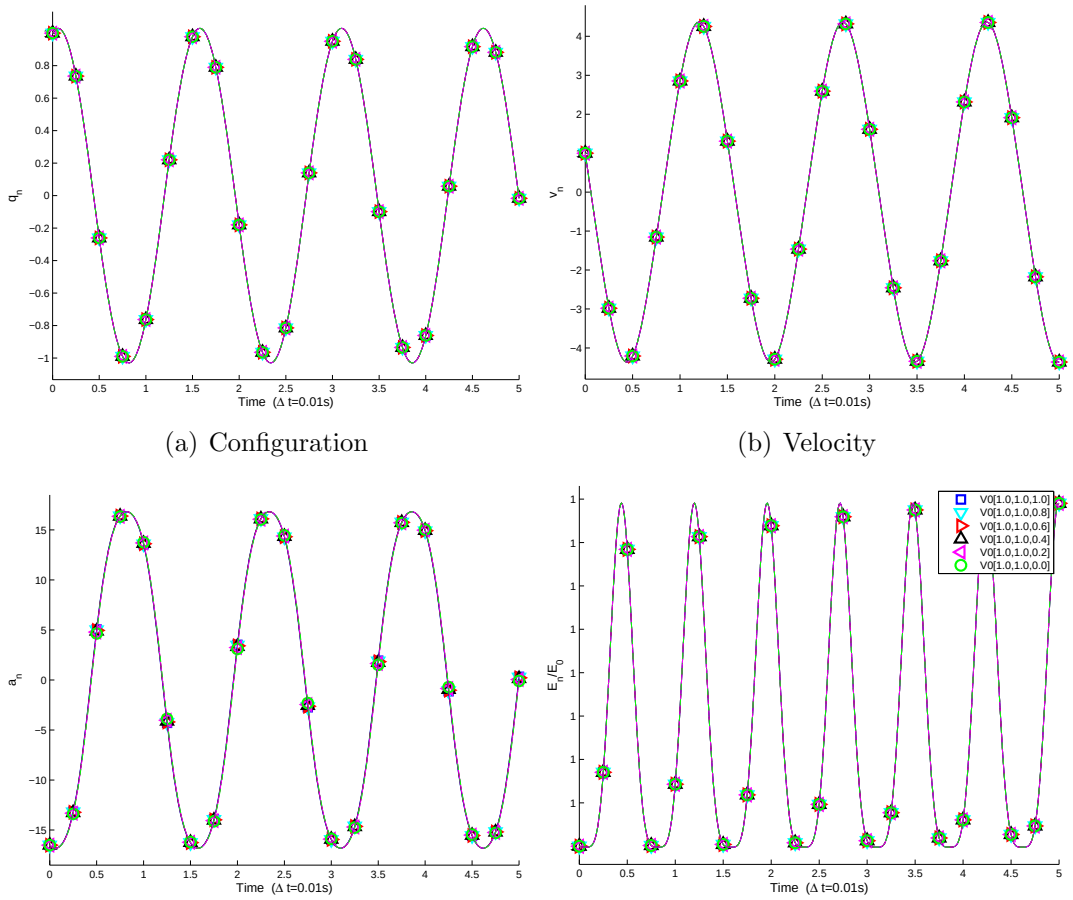
Figure A.54: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Simple pendulum] [Implicit GSSSS family of algorithms (Option II) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

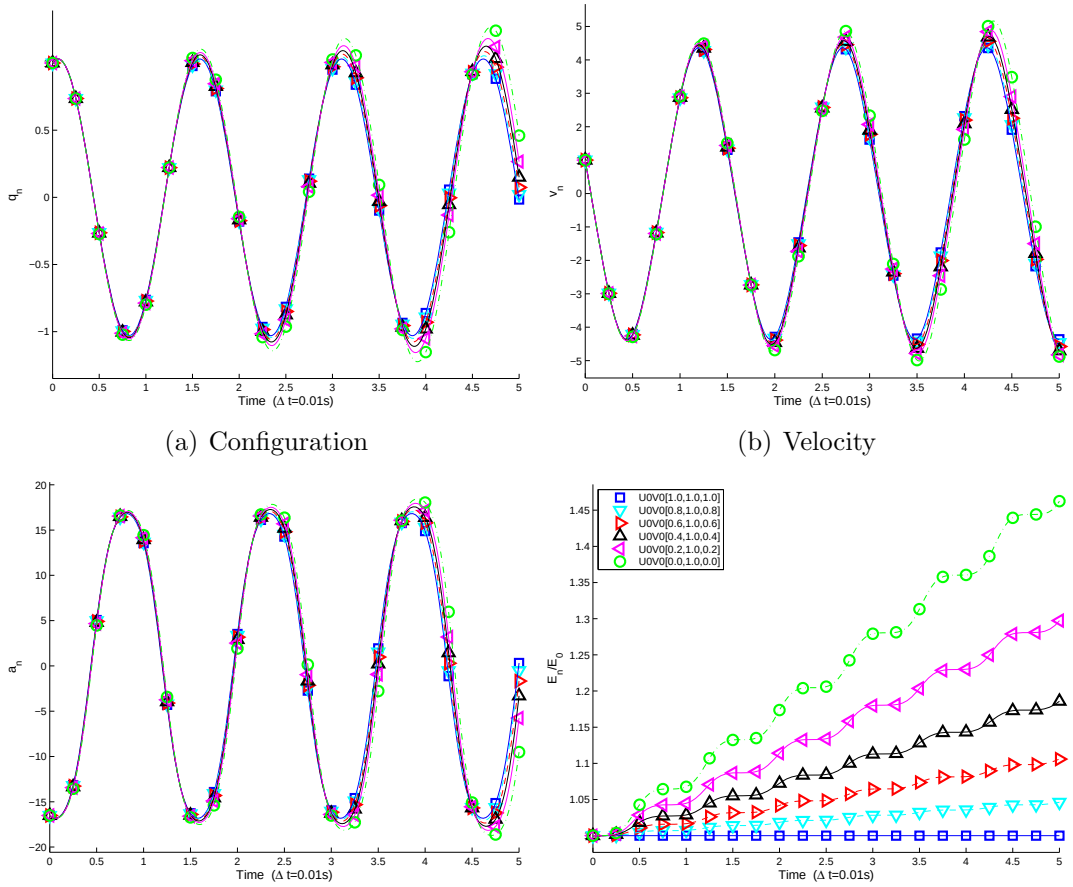
Figure A.55: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Simple pendulum] [Implicit GSSSS family of algorithms (Option II) - U0V0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

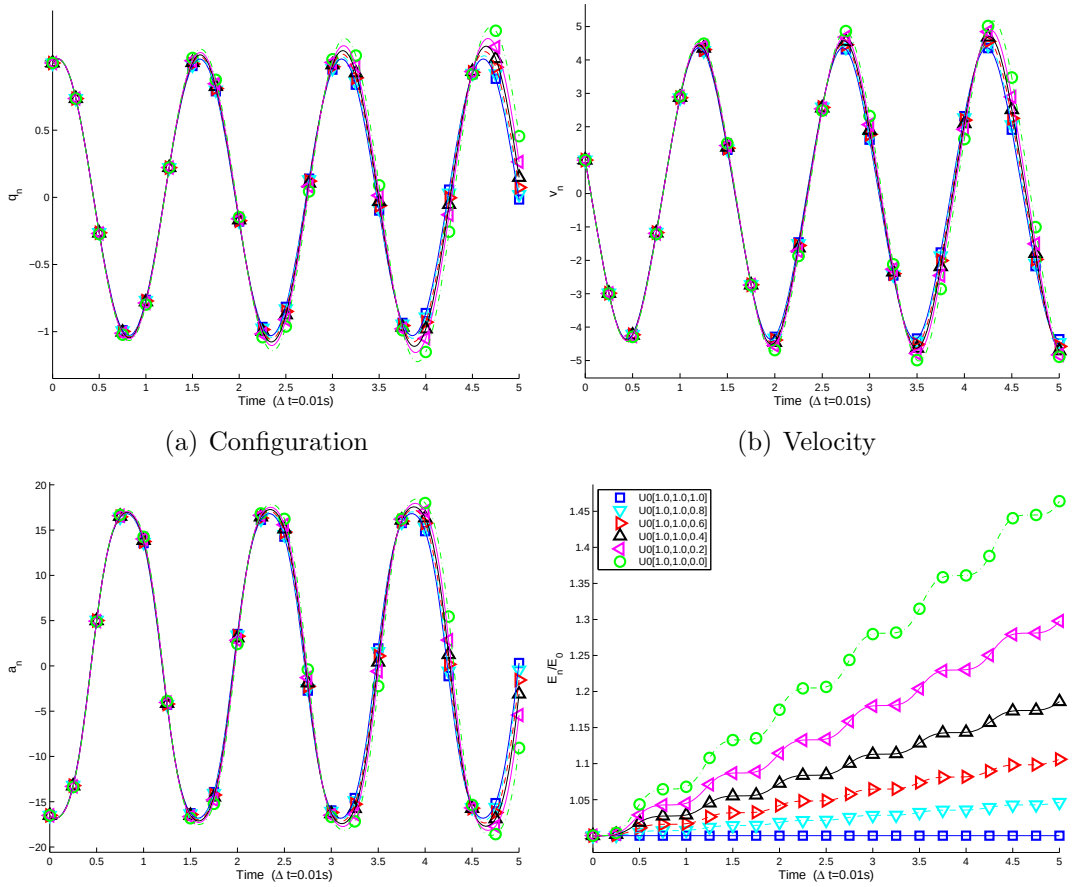
Figure A.56: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Simple pendulum] [Implicit GSSSS family of algorithms (Option II) - V0U0(1.0,1.0,$\rho_{\infty}^{\mathrm{s}}$)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

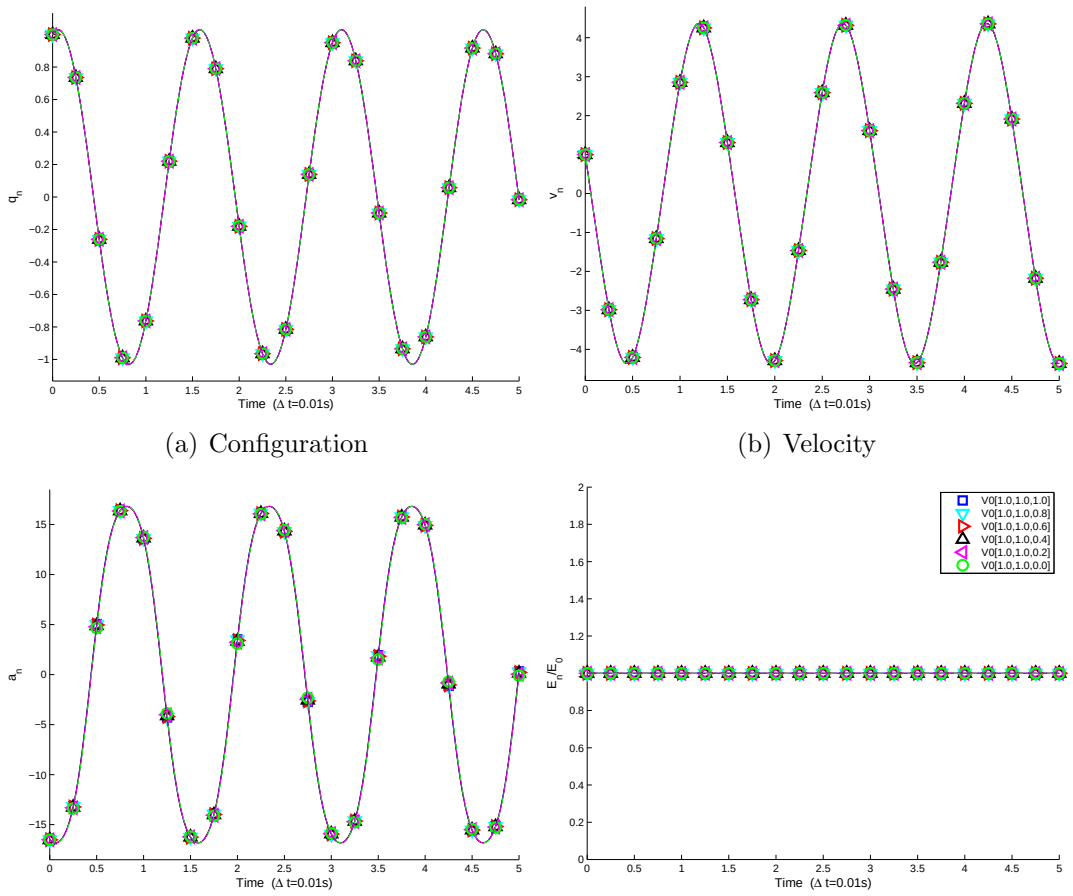
Figure A.57: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Simple pendulum] [Implicit GSSSS family of algorithms (Option III) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

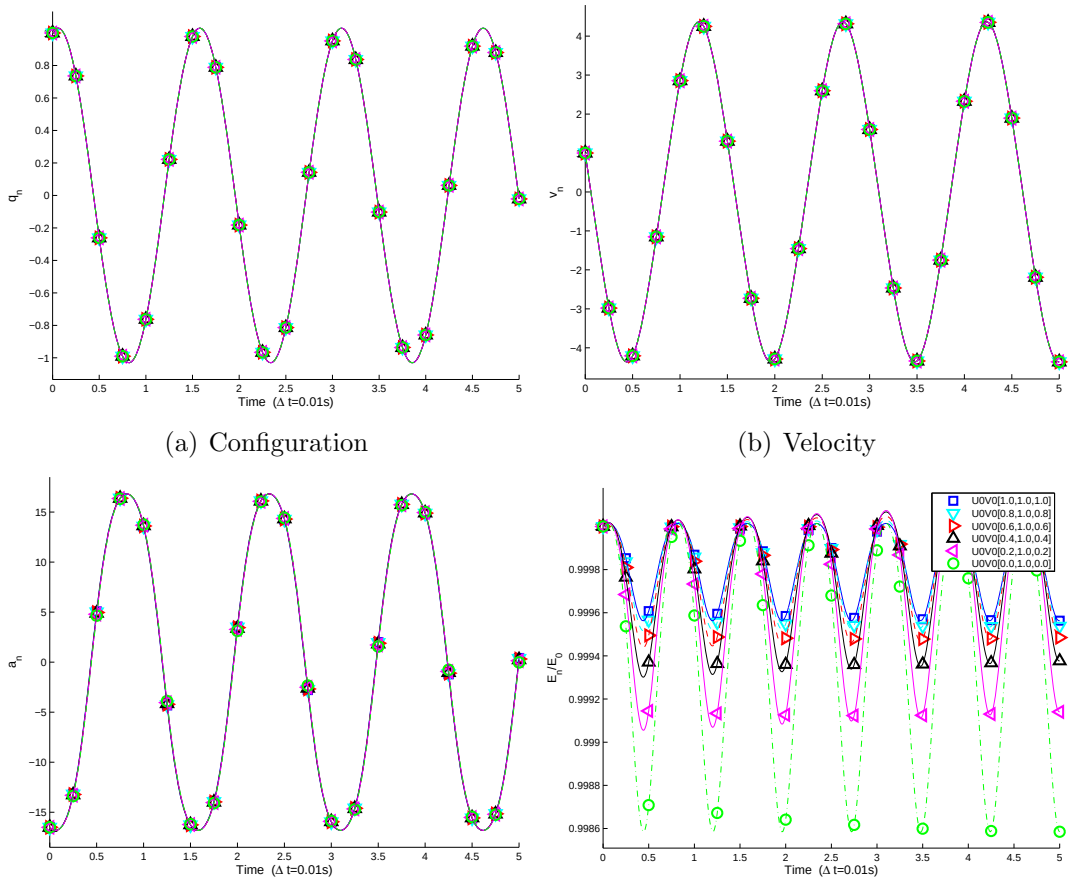
Figure A.58: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Simple pendulum] [Implicit GSSSS family of algorithms (Option III) - U0V0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

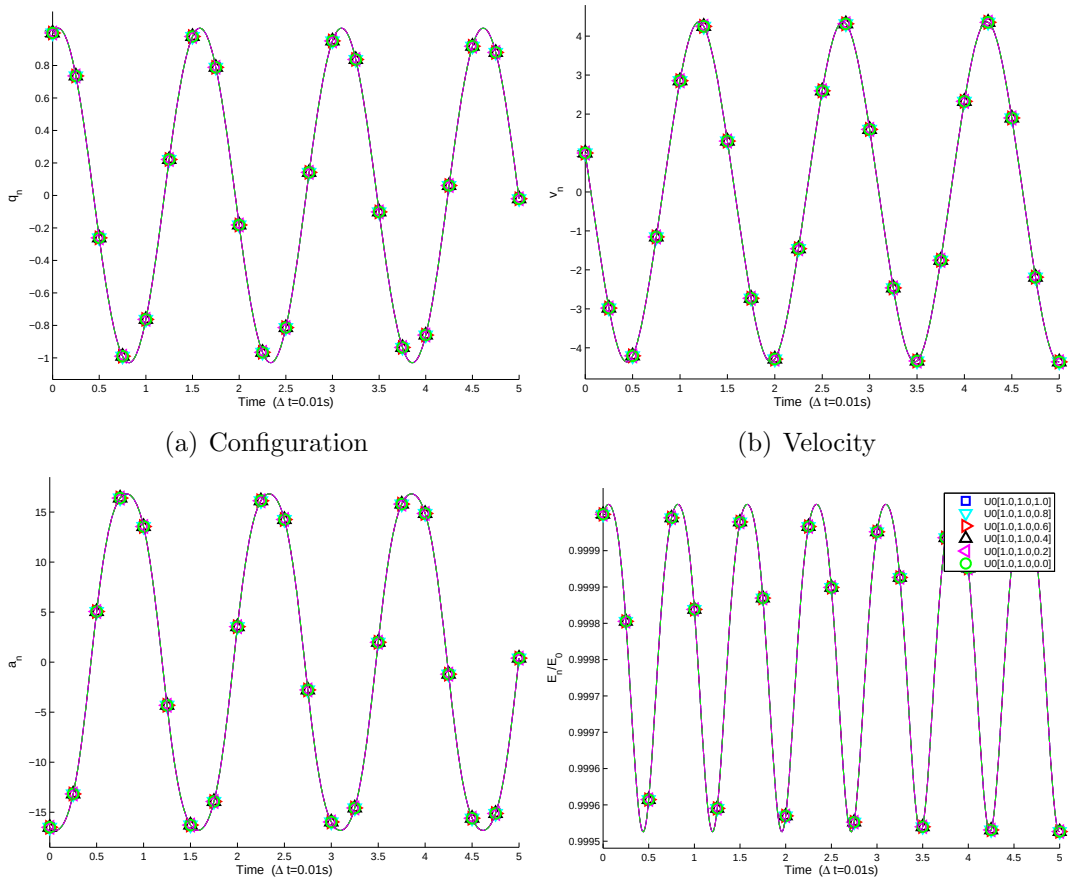
Figure A.59: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Simple pendulum] [Implicit GSSSS family of algorithms (Option III) - V0U0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

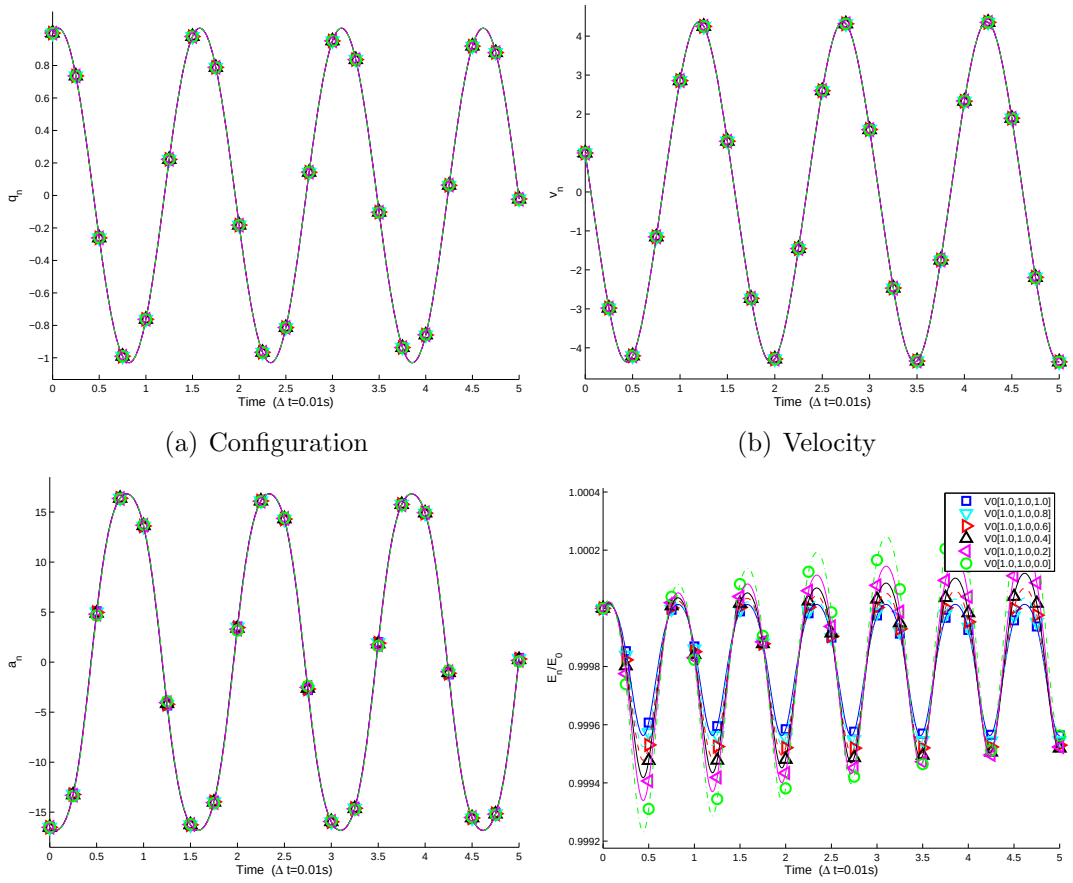
Figure A.60: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Simple pendulum] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option I) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

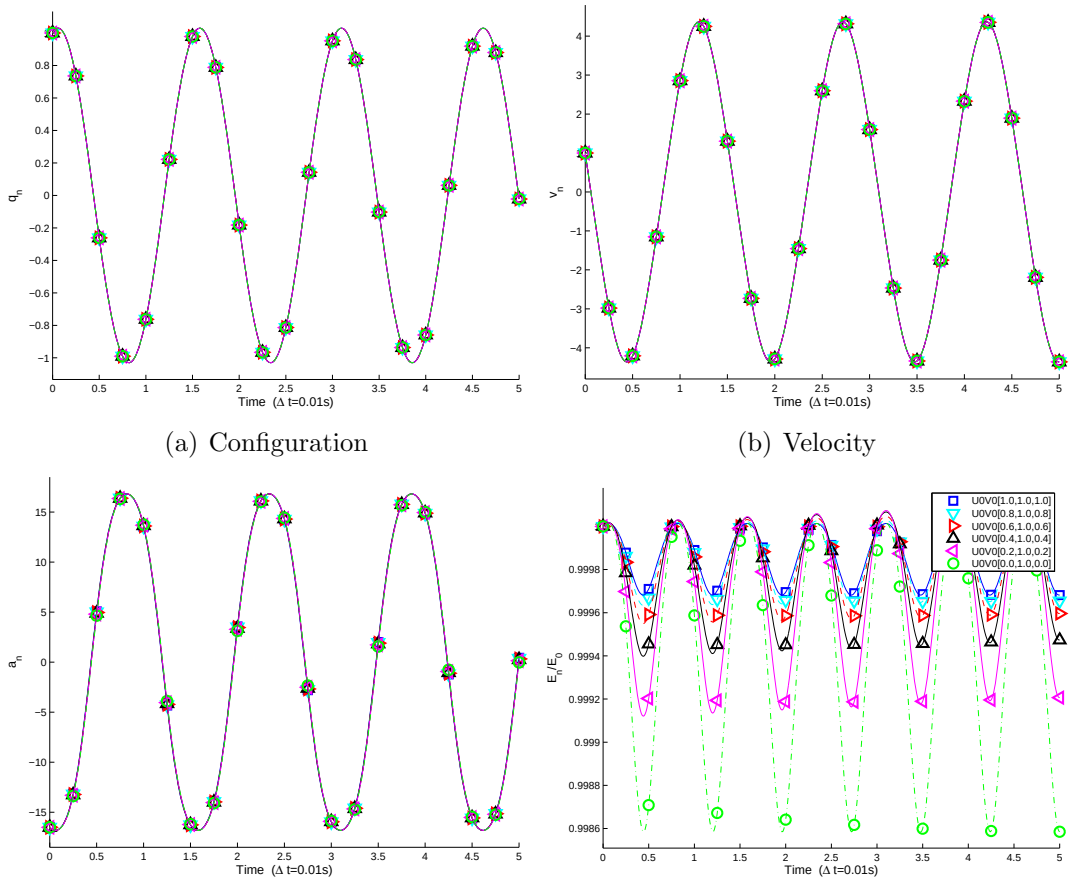
Figure A.61: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Simple pendulum] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option I) - U0V0(1.0,1.0,$\rho^{\mathrm{s}}_{\infty}$)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

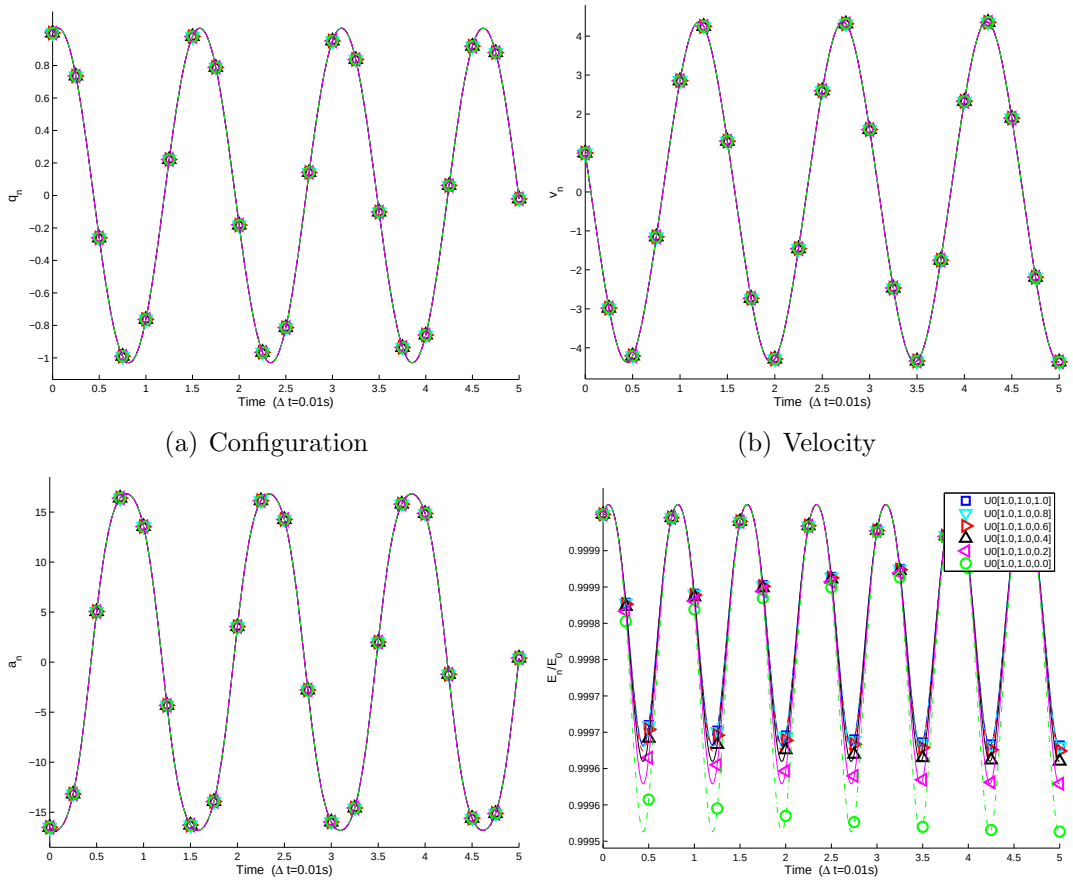
Figure A.62: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Simple pendulum] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option I) - V0U0(1.0,1.0,ρ^s_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

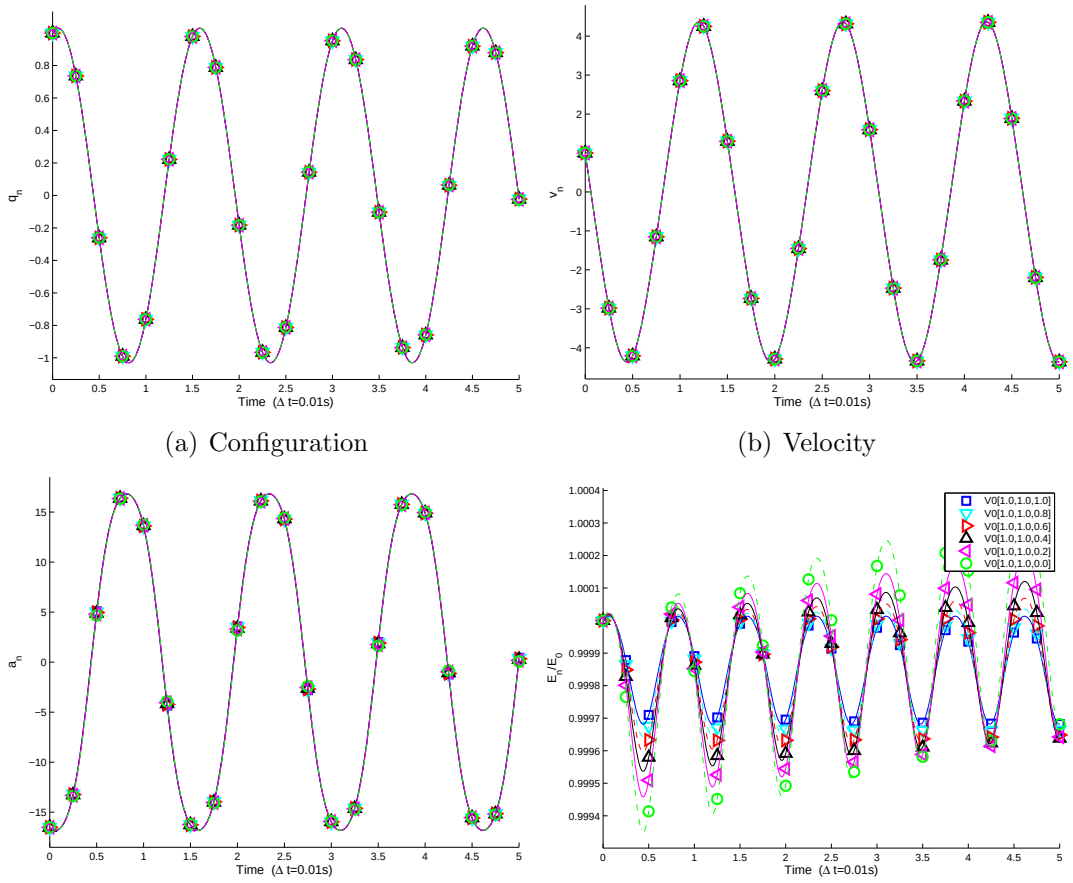
Figure A.63: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Simple pendulum] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option II) - U0V0/V0U0(ρ_∞,1.0,ρ_∞)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

Figure A.64: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Simple pendulum] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option II) - U0V0(1.0,1.0,ρ_∞^s)]



(a) Configuration (b) Velocity

(c) Acceleration (d) Total Energy Ratio

Figure A.65: Time histories of the configuration, velocity, acceleration, and the ratio of total energy in the ***conservative system*** with $\Delta t = 0.01$. [Problem: Simple pendulum] [PCE-GSSSS ($\eta_3 = 0$) family of algorithms (Option II) - V0U0(1.0,1.0,ρ_∞^s)]

A.10 Numerical Results for the Hyperelastic Elastodynamics Nonlinear Problems

As an example of nonlinear hyperelastic material models, we choose the neo-hookean material model. It is one of the simplest material model, and its stored energy is defined by

$$\hat{W}^h = \frac{\mu}{2}(\mathrm{tr}(\mathbf{C}^h) - 3) - \mu \ln(J) + \frac{\lambda}{2}\ln(J)^2 \tag{A.60}$$

where $\mu > 0$ and $\lambda > 0$ are the Lame parameters, $\mathbf{C}^h$ is the discrete right Cauchy-Green tensor, and $J := \det(\mathbf{F}^h)$. These parameters can be expressed in terms of Young's modulus E and Poisson's ratio ν by

$$\begin{aligned} \lambda &= \frac{E\nu}{(1+\nu)(1-2\nu)} \\ \mu &= \frac{E}{2(1+\mu)} \end{aligned} \tag{A.61}$$

The numerical results for the hyperelastic elastodynamics problems, illustrated in Fig. A.10, with the Neo-Hookean material model, using the implicit GSSSS family of algorithms and the PCE GSSSS family of algorithms, are shown below. Note that all schemes are second-order time accurate; see Fig. A.67 - Fig. A.69. And, the mechanical energy of the system is exactly conserved when $W_1 = 1/2$ in the framework of the modified Option III GSSSS family of algorithms; otherwise, it oscillates or dissipates; see Fig. A.70 - Fig. A.73. When $U0/V0(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}}) = U0/V0(1,1,1)$ or $V0(\rho_\infty^{\min}, \rho_\infty^{\max}, \rho_\infty^{\mathrm{s}}) = V0(1,1,0)$ is selected, the mechanical energy is exactly conserved in the modified Option III implicit GSSSS family of schemes as expected. When $W_1 = 1/2$ in the modified Option III GSSSS family of algorithms, the discrete algorithmic second Piola Kirchhoff stress tensor is given by

$$\mathbf{S}^h_{\mathrm{alg}} = 2D\hat{W}(\tilde{\mathbf{C}}^h) + 2\frac{\hat{W}(\mathbf{C}^h_{n+1}) - \hat{W}(\mathbf{C}^h_n) - D\hat{W}(\tilde{\mathbf{C}}^h) : \Delta\mathbf{C}^h}{\| \Delta\mathbf{C}^h \|^2}\Delta\mathbf{C}^h \tag{A.62}$$

otherwise,

$$\mathbf{S}^h_{\mathrm{alg}} = 2D\hat{W}(\tilde{\mathbf{C}}^h) \tag{A.63}$$

(see Algorithm 27).

For this example problem, we applied the following three families of algorithms, i.e., the implicit GSSSS family of algorithms in the sense of Option II and modified Option III, and some selected variational explicit schemes within the PCE-GSSSS family of algorithms in the sense of Options II and III. The selected variational schemes are: U0(1.0,1.0,1.0), U0(0.0,1.0,0.0), U0(1.0,1.0,0.0), U0(1,1,0.5), U0(0.5.1.0), and U0(0.0,0.0,0.0) algorithms. Note that the implicit GSSSS family of algorithms (Option II) and GSSSS family of algorithms (Modified Option III) are the variational-based and the exact energy-momentum conserving-based family of schemes. For each implicit algorithm, we select,

U0V0/V0U0(ρ_∞,1.0,ρ_∞)

U0V0(1.0,1.0,ρ_∞^{s})

V0U0(1.0,1.0,ρ_∞^{s})

where

$$\begin{aligned} &\rho_\infty = \rho_\infty^{\min} = \rho_\infty^{\max} \in \{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\} \\ &\rho_\infty^{\mathrm{s}} \in \{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\} \end{aligned} \tag{A.64}$$

The enlarged legends of the figures are shown in Fig. 4.5. As expected, we have the second-order time accuracy in the configuration, velocity, and acceleration vectors for all implicit and explicit schemes; see Fig. A.67 - Fig. A.69. In Fig. A.70 - Fig. A.73, the responses of the mechanical energy and the angular momentum for the selected implicit and explicit schemes are shown. The plots are shown from $t = 0.1$ after $k = 10$ first time steps (the time step sized used for the simulation is $\Delta t = 0.01$). There exists no exact energy conserving scheme within the implicit GSSSS family of algorithms (Option II); however, we can obtain better energy conserving features by selecting higher spectral values. On the other hand, the mechanical energy is exactly conserved within the time step by selecting $\rho_\infty^{\min} = \rho_\infty^{\max} = \rho_\infty^{\mathrm{s}} = 1$ in the U0-based family or $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\mathrm{s}} \in [0, 1]$ in the V0-based family within the implicit GSSSS family of algorithms (Modified Option III). We can conserve the angular momentum exactly in the sense of $\mathbf{J}_n = \mathbf{J}_{n+1}$ in the conservative system in the implicit GSSSS family of algorithms (both for

Option II and modified Option III) by selecting $\rho_\infty^{\min} = \rho_\infty^{\max} = \rho_\infty^{\mathrm{s}} = 1$ in the U0-based family or $\rho_\infty^{\min} = \rho_\infty^{\max} = 1$ and $\rho_\infty^{\mathrm{s}} \in [0, 1]$ in the V0-based family. The selected variational explicit GSSSS algorithms show that the mechanical energy and angular momentum does not dissipate, but it is bounded. Note that only U0(1,1,0) scheme conserves the angular momentum within the time step exactly; otherwise, it is bounded.

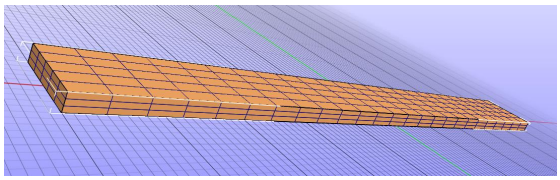
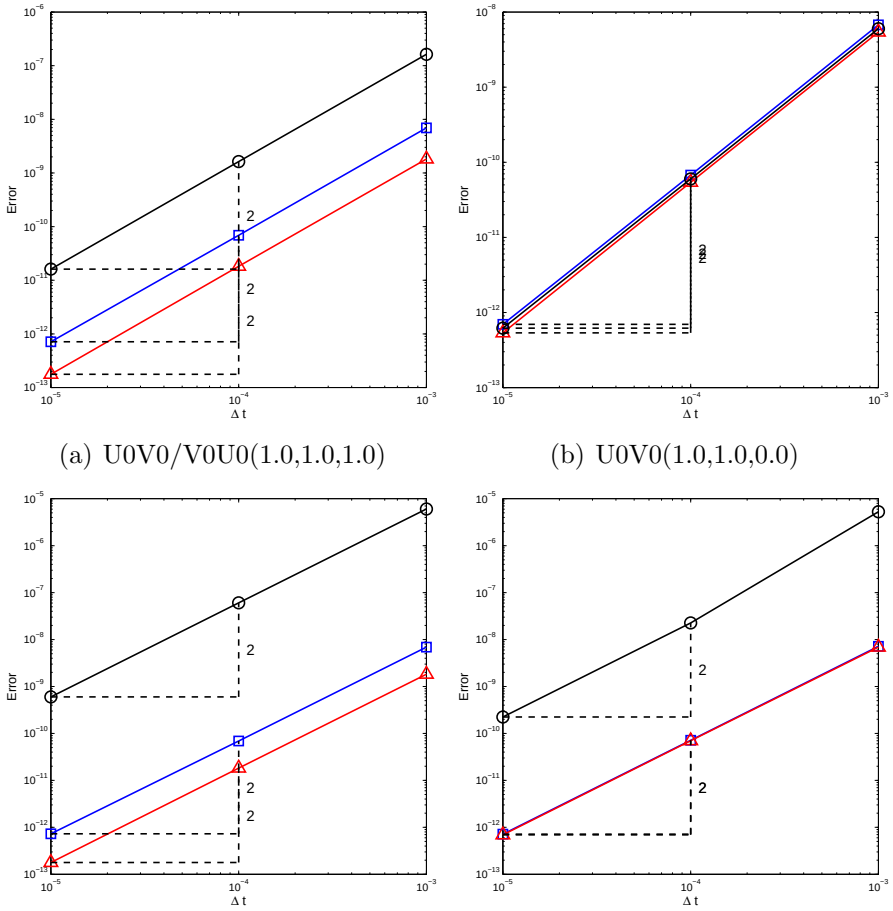


Figure A.66: Geometry and finite element mesh for a 3-D block. Input parameters are: Number of elements = 375, $\rho = 1.0$ (density), $E = 10^6$, $\nu = 0.3$, $\mathbf{F} = (0.0, -0.5, 0.0)$ for $0 \leq t < 0.05$.



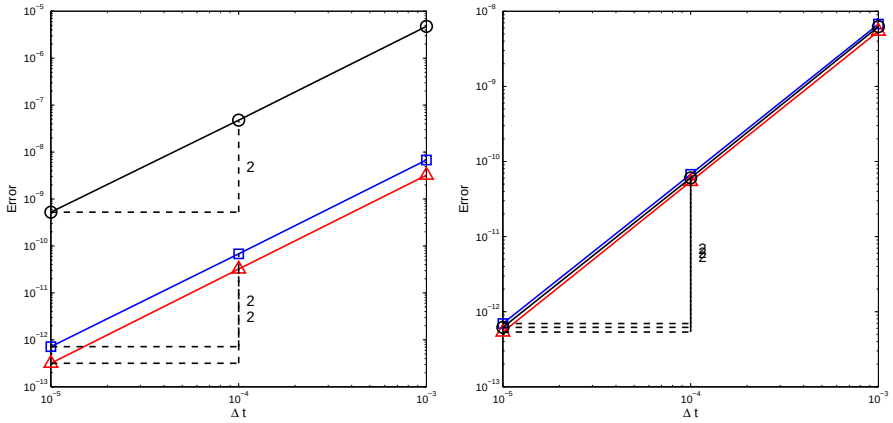
(a) U0V0/V0U0(1.0,1.0,1.0)

(b) U0V0(1.0,1.0,0.0)

(c) V0U0(1.0,1.0,0.0)

(d) U0V0/V0U0(0.8,1.0,0.8)

Figure A.67: Time accuracies in the configuration, velocity, and acceleration. [Problem: 3D Nonlinear Bar (Neo-Hookean)] [Implicit GSSSS algorithms (Option II)]



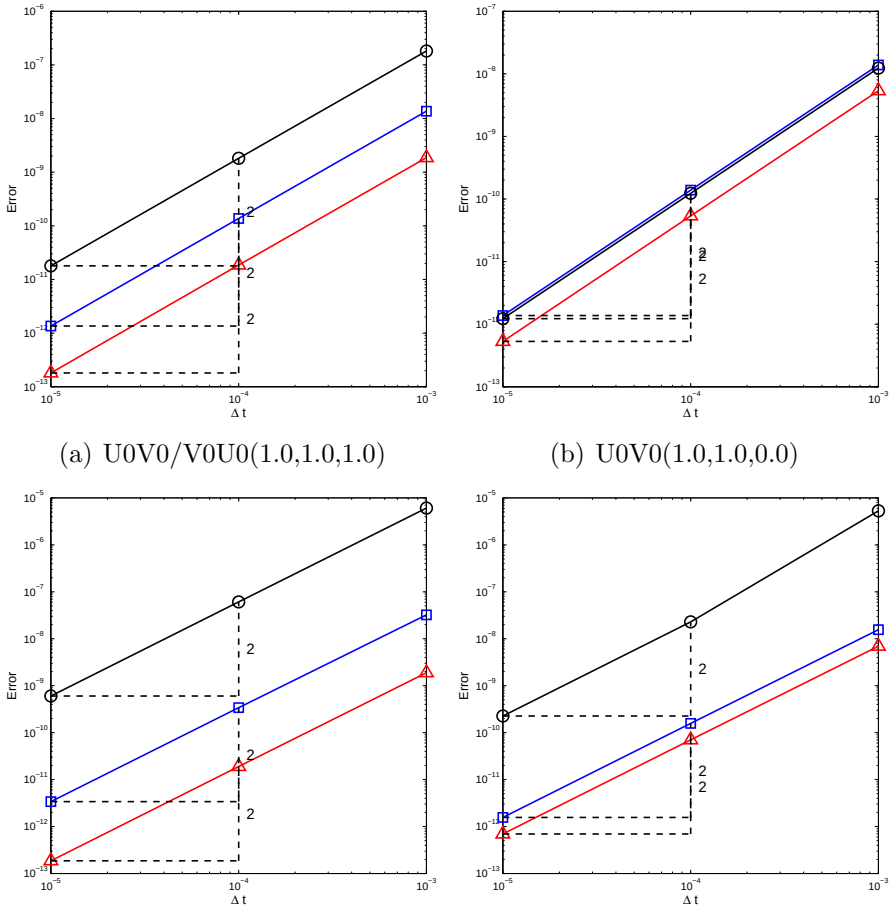
(a) U0V0/V0U0(1.0,1.0,1.0)

(b) U0V0(1.0,1.0,0.0)

(c) V0U0(1.0,1.0,0.0)

(d) U0V0/V0U0(0.8,1.0,0.8)

Figure A.68: Time accuracies in the configuration, velocity, and acceleration. [Problem: 3D Nonlinear Bar (Neo-Hookean)] [Implicit GSSSS algorithms (Modified Option III)]



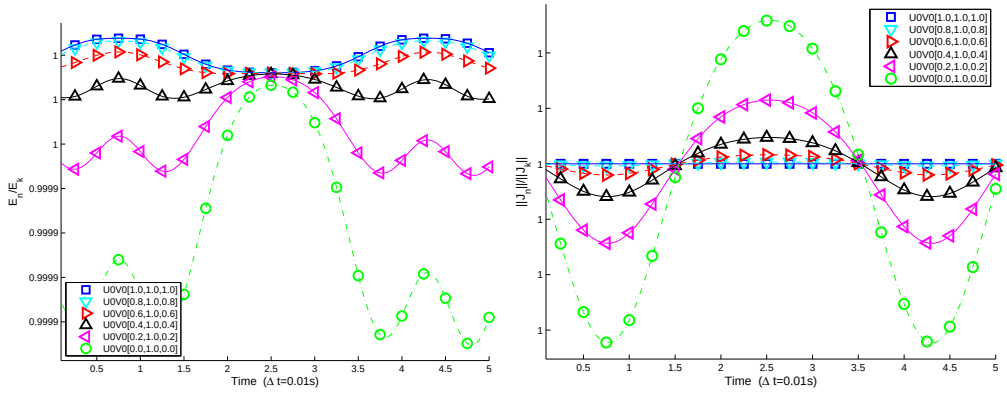
(a) U0V0/V0U0(1.0,1.0,1.0)

(b) U0V0(1.0,1.0,0.0)

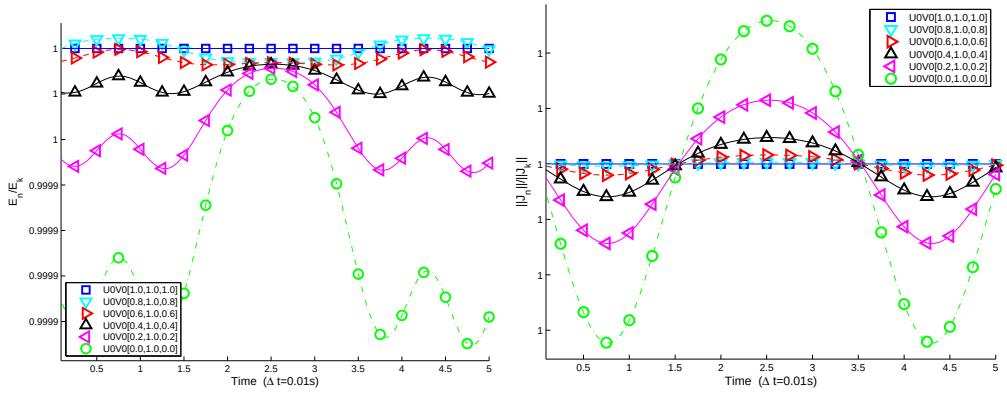
(c) V0U0(1.0,1.0,0.0)

(d) U0V0/V0U0(0.8,1.0,0.8)

Figure A.69: Time accuracies in the configuration, velocity, and acceleration. [Problem: 3D Nonlinear Bar (Neo-Hookean)] [PCE-GSSSS algorithms ($\eta_3 = 0$) (Option II)]

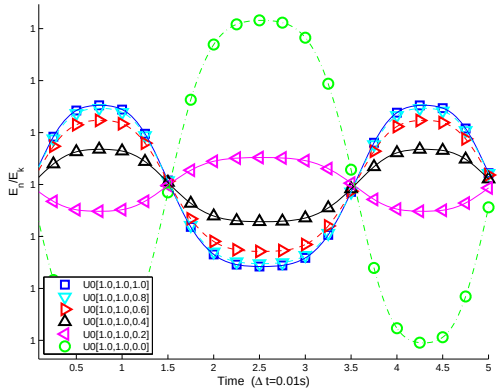


(a) Option II: Mechanical Energy Ratio (b) Option II: Angular Momentum Ratio

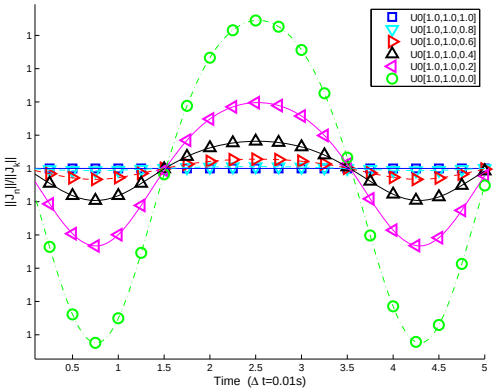


(c) Option III (Modified): Mechanical Energy Ratio (d) Option III (Modified): Angular Momentum Ratio

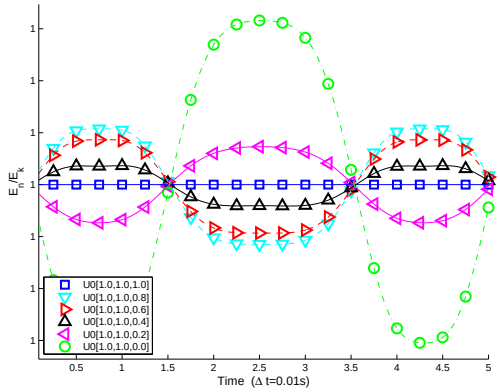
Figure A.70: Mechanical energy and angular momentum (Ratio) history for 3D Nonlinear block (Neo-Hookean): Implicit U0V0 Optimal Schemes ($k = 10$)



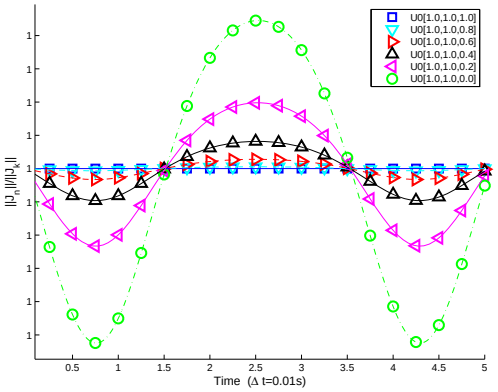
(a) Option II: Mechanical Energy Ratio



(b) Option II: Angular Momentum Ratio

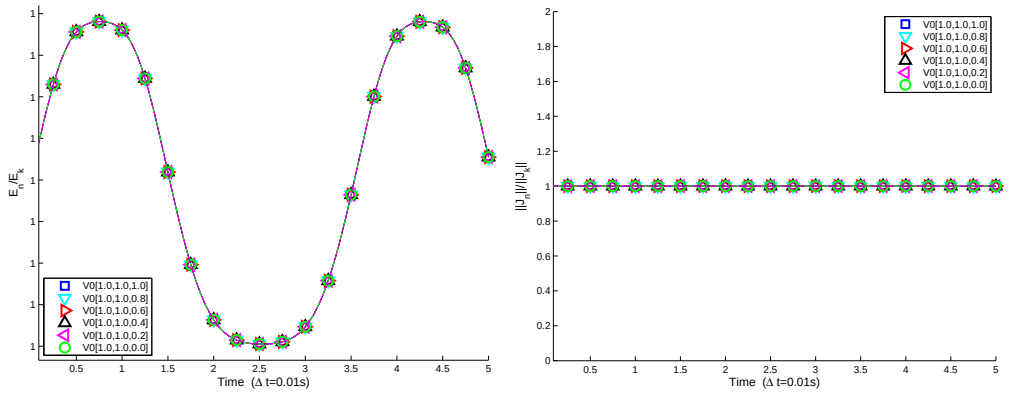


(c) Option III (Modified): Mechanical Energy Ratio



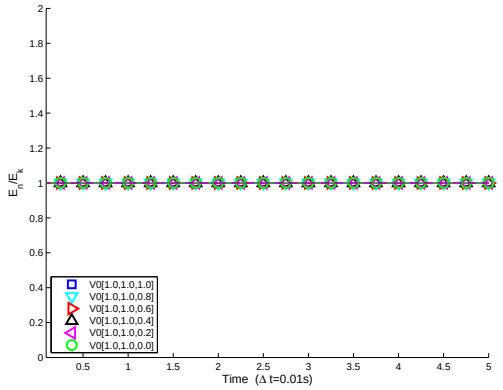
(d) Option III (Modified): Angular Momentum Ratio

Figure A.71: Mechanical energy and angular momentum (Ratio) history for 3D Nonlinear block (Neo-Hookean): Implicit U0V0-based Schemes ($k = 10$)

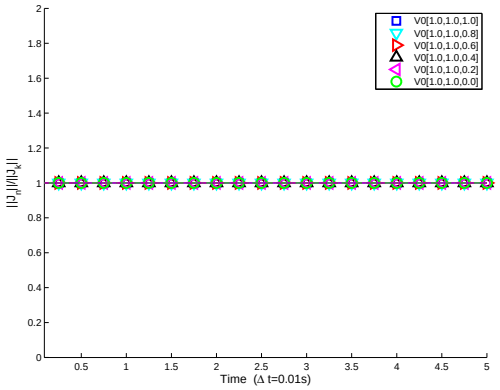


(a) Option II: Mechanical Energy Ratio

(b) Option II: Angular Momentum Ratio

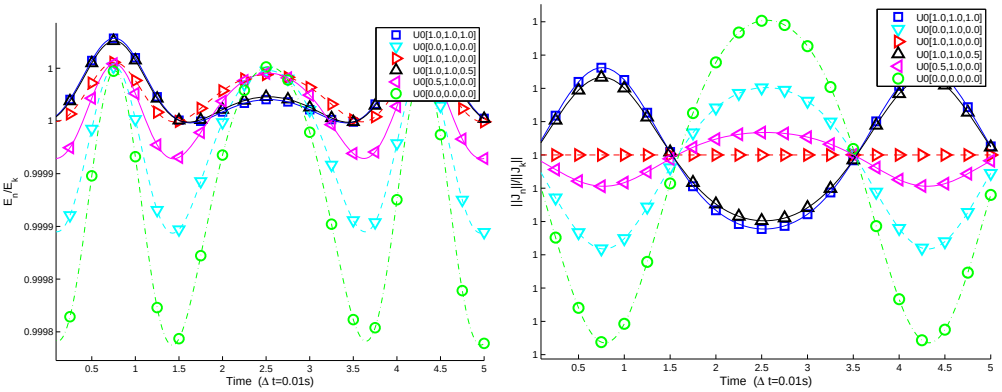


(c) Option III (Modified): Mechanical Energy Ratio

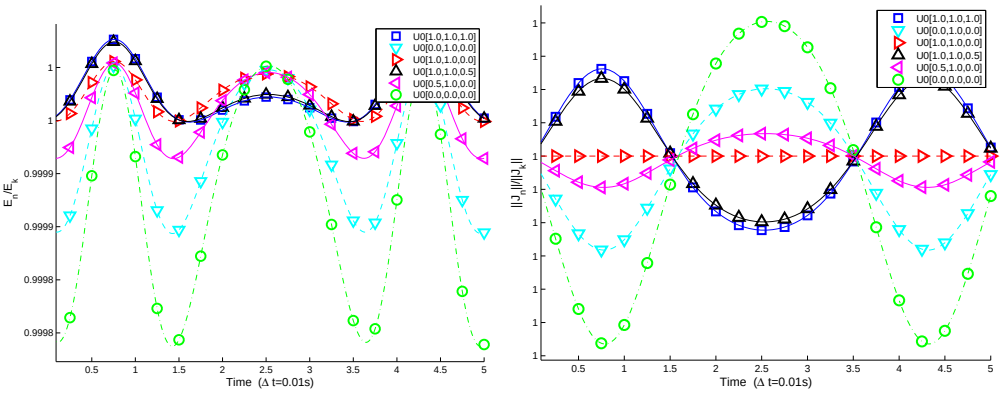


(d) Option III (Modified): Angular Momentum Ratio

Figure A.72: Mechanical energy and angular momentum (Ratio) history for 3D Nonlinear block (Neo-Hookean): Implicit V0U0-based Schemes ($k = 10$)



(a) Option II: Mechanical Energy Ratio (b) Option II: Angular Momentum Ratio



(c) Option III: Mechanical Energy Ratio (d) Option III: Angular Momentum Ratio

Figure A.73: Mechanical energy and angular momentum (Ratio) history for 3D Nonlinear block (Neo-Hookean): Selected PCE-GSSSS Schemes in Option II and Option III ($\eta_3 = 0$) ($k = 10$)

Appendix B

Literature Review: Energy Momentum Conserving Algorithms for N-body Systems and Elastodynamics

B.1 Energy-Momentum Conserving Algorithm for N-Body System (Simo-Tarnow-Wong Framework)

In this section, we summarize the pioneering original work of constructing the exact energy-momentum conserving algorithm for N-body system, described in Simo *et.al.* [52]. Assume the autonomous Hamiltonian $\mathscr{H} : T^*Q \to \mathbb{R}$ to be completely separable; namely, $\mathscr{H}(\boldsymbol{q}, \boldsymbol{p}) = \mathscr{T}(\boldsymbol{p}) + \mathscr{U}(\boldsymbol{q})$, where the kinetic energy and the potential energy of the system are $\mathscr{T} : T^*Q \to \mathbb{R}$ and $\mathscr{U} : Q \to \mathbb{R}$, respectively. Recall that the Hamiltonian is defined on the cotangent bundle T^*Q ($\equiv P$, phase space). Assume the configuration manifold Q of a finite dimensional system is a linear space. Consider the case where the kinetic energy is a positive

definite quadratic form in the canonical momenta, i.e.,

$$\mathscr{T}(\boldsymbol{p}) = \frac{1}{2}\boldsymbol{p}\cdot\boldsymbol{M}^{-1}\boldsymbol{p} \geq 0 \tag{B.1}$$

where the symmetric mass matrix $\boldsymbol{M}$ is *constant*, and a set of the generalized momenta, $\boldsymbol{p} = (\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_N) \in T_q^* Q \equiv \mathbb{R}^{3N}$ is defined with the Lagrangian $\mathscr{L}$ by

$$\boldsymbol{p}_i = \frac{\partial \mathscr{L}}{\partial \dot{\boldsymbol{q}}^i} \tag{B.2}$$

for $i = 1, 2, \ldots, N$. Recall that the Lagrangian is defined in the tangent bundle TQ ($\equiv S$, velocity phase space), and the transformation from the (regular) Lagrangian to the Hamiltonian is achieved via the Legendre transformation: $(\boldsymbol{q}, \dot{\boldsymbol{q}}) \in TQ \mapsto (\boldsymbol{q}, \boldsymbol{p}) \in T^*Q$; $\mathscr{L}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \mapsto \mathscr{H}(\boldsymbol{q}, \boldsymbol{p}) = \sum_{i=1}^{N} \boldsymbol{p}_i \cdot \boldsymbol{\psi}^i - \mathscr{L}(\boldsymbol{q}, \dot{\boldsymbol{q}})$, where $\dot{\boldsymbol{q}} = \boldsymbol{\psi}(\boldsymbol{q}, \boldsymbol{p}) \in T_q Q$ obtained from equation (B.2). In this special case where the kinetic energy is given as the form of equation (B.1) and the potential energy depends only upon a set of generalized coordinates $\boldsymbol{q} = (\boldsymbol{q}^1, \boldsymbol{q}^2, \ldots, \boldsymbol{q}^N) \in Q \equiv \mathbb{R}^{3N}$ explicitly, the Hamiltonian is identical to the total energy of the system. The equation of motion of the system is described by the following Hamilton's canonical equations:

$$\dot{\boldsymbol{q}} = \frac{\partial \mathscr{H}(\boldsymbol{q}, \boldsymbol{p})}{\partial \boldsymbol{p}} = \boldsymbol{M}^{-1}\boldsymbol{p} \tag{B.3a}$$

$$\dot{\boldsymbol{p}} = -\frac{\partial \mathscr{H}(\boldsymbol{q}, \boldsymbol{p})}{\partial \boldsymbol{q}} = -\nabla \mathscr{U}(\boldsymbol{q}) \tag{B.3b}$$

Consider a time interval $\mathbb{I} = [t_0, T]$ split into subintervals, i.e., $\mathbb{I} = [t_0, T] = \bigcup_{n=0}^{k-1} [t_n, t_{n+1}]$ where $t_k = T$, and the focus is upon a typical time interval $[t_n, t_{n+1}] \subset \mathbb{I}$ in the following time discretization. Based upon the invariant properties of the autonomous Hamiltonian system, developing a consistent algorithmic time integration scheme satisfying the conservation of the linear momentum $\boldsymbol{L}$, angular momentum $\boldsymbol{J}$, and the total energy $\mathscr{H}$ in the typical time interval as show below is desired:

$$\boldsymbol{L}_n = \boldsymbol{L}_{n+1}, \quad \boldsymbol{J}_n = \boldsymbol{J}_{n+1}, \quad \mathscr{H}_n = \mathscr{H}_{n+1} \tag{B.4}$$

where $\boldsymbol{L}_n := \boldsymbol{L}(\boldsymbol{p}_n)$ and $\boldsymbol{L}_{n+1} := \boldsymbol{L}(\boldsymbol{p}_{n+1})$; $\boldsymbol{J}_n := \boldsymbol{J}(\boldsymbol{q}_n, \boldsymbol{p}_n)$ and $\boldsymbol{J}_{n+1} := \boldsymbol{J}(\boldsymbol{q}_{n+1}, \boldsymbol{p}_{n+1})$; $\mathscr{H}_n := \mathscr{H}(\boldsymbol{q}_n, \boldsymbol{p}_n)$ and $\mathscr{H}_{n+1} := \mathscr{H}(\boldsymbol{q}_{n+1}, \boldsymbol{p}_{n+1})$.

Next, define the convex combinations for the canonical coordinates and the canonical momenta as

$$\boldsymbol{q}_{n+\alpha} := (1-\alpha)\,\boldsymbol{q}_n + \alpha \boldsymbol{q}_{n+1} \in Q \equiv \mathbb{R}^{3N} \tag{B.5}$$

$$\boldsymbol{p}_{n+(1-\alpha)} := \alpha \boldsymbol{p}_n + (1-\alpha)\,\boldsymbol{p}_{n+1} \in T_q^* Q \equiv \mathbb{R}^{3N} \tag{B.6}$$

for $\alpha \in [0,1]$, respectively. The translational invariance of the Hamiltonian leads to conservation of the linear momentum, i.e.,

$$\frac{\partial \mathscr{H}(\boldsymbol{q}, \boldsymbol{p})}{\partial \boldsymbol{q}} = \nabla \mathscr{U}(\boldsymbol{q}) = \dot{\boldsymbol{p}} = \boldsymbol{0} \tag{B.7}$$

Thus, in view of equation (B.3), the linear momentum is automatically conserved in the absence of conservative loadings. The linear momentum is conserved regardless of the type of a time integration scheme by the dynamics.

Assuming that the kinetic energy $\mathscr{T}(\boldsymbol{p})$ and potential energy $\mathscr{U}(\boldsymbol{q})$ are $SO(3)$-frame invariant, i.e.,

$$\mathscr{T}(\boldsymbol{Q} \circ \boldsymbol{p}) := \mathscr{T}(\boldsymbol{Q}\boldsymbol{p}_1, \boldsymbol{Q}\boldsymbol{p}_2, \ldots, \boldsymbol{Q}\boldsymbol{p}_N) \equiv \mathscr{T}(\boldsymbol{p}) \tag{B.8}$$

$$\mathscr{U}(\boldsymbol{Q} \circ \boldsymbol{q}) := \mathscr{U}(\boldsymbol{Q}\boldsymbol{q}^1, \boldsymbol{Q}\boldsymbol{q}^2, \ldots, \boldsymbol{Q}\boldsymbol{q}^N) \equiv \mathscr{U}(\boldsymbol{q}) \tag{B.9}$$

$\forall \boldsymbol{Q} \in SO(3)$, we can show that the Hamiltonian is also $SO(3)$-frame invariant,

$$\mathscr{H}(\boldsymbol{Q} \circ \boldsymbol{q}, \boldsymbol{Q} \circ \boldsymbol{p}) \equiv \mathscr{H}(\boldsymbol{q}, \boldsymbol{p}) \tag{B.10}$$

For the satisfaction of the $SO(3)$-frame invariance of the Hamiltonian and the potential energy, the following are required:

$$\sum_{A=1}^{N} \left[\boldsymbol{q}^A \times \frac{\partial \mathscr{H}}{\partial \boldsymbol{q}^A} + \boldsymbol{p}_A \times \frac{\partial \mathscr{H}}{\partial \boldsymbol{p}_A} \right] = \boldsymbol{0} \tag{B.11}$$

$$\sum_{A=1}^{N} \left[\boldsymbol{q}^A \times \frac{\partial \mathscr{U}}{\partial \boldsymbol{q}^A} \right] = \boldsymbol{0} \tag{B.12}$$

respectively. Notice $\boldsymbol{p}_A$ is the covariant vectors; and $\boldsymbol{q}^A$ are neither the covariant nor contravariant vectors, but since their derivatives $d\boldsymbol{q}^A$ are contravariant vectors, the superscript A is used. Directly from the definition of the angular momentum $\boldsymbol{J} : T^*Q \to \mathbb{R}^3$; that is,

$$\boldsymbol{J}(\boldsymbol{q},\boldsymbol{p}) := \sum_{A=1}^{N} \boldsymbol{q}^A \times \boldsymbol{p}_A \tag{B.13}$$

the finite increment of the angular momentum within a time step $[t_n, t_{n+1}]$ can be written as

$$\boldsymbol{J}_{n+1} - \boldsymbol{J}_n = \sum_{A=1}^{N} \left[\boldsymbol{q}^A_{n+\alpha} \times \left(\boldsymbol{p}_{An+1} - \boldsymbol{p}_{An} \right) - \boldsymbol{p}_{An+(1-\alpha)} \times \left(\boldsymbol{q}^A_{n+1} - \boldsymbol{q}^A_n \right) \right] \tag{B.14}$$

A comparison between equation (B.14) and the following relations obtained from equation (B.11) and equation (B.12) with the convex conbinations $\boldsymbol{q}_{n+\alpha}$ and $\boldsymbol{p}_{n+\alpha}$

$$\sum_{A=1}^{N} \boldsymbol{q}^A_{n+\alpha} \times \frac{\partial \mathscr{U}\left(\boldsymbol{q}_{n+\alpha}\right)}{\partial \boldsymbol{q}^A_{n+\alpha}} = \boldsymbol{0} \tag{B.15}$$

$$\sum_{A=1}^{N} \boldsymbol{p}_{An+(1-\alpha)} \times \frac{\partial \mathscr{T}\left(\boldsymbol{p}_{n+(1-\alpha)}\right)}{\partial \boldsymbol{p}_{An+(1-\alpha)}} = \boldsymbol{0} \tag{B.16}$$

yields

$$\boldsymbol{q}_{n+1} - \boldsymbol{q}_n \propto \frac{\partial \mathscr{T}\left(\boldsymbol{p}_{n+(1-\alpha)}\right)}{\partial \boldsymbol{p}_{n+(1-\alpha)}} = \boldsymbol{M}^{-1} \boldsymbol{p}_{n+(1-\alpha)} \tag{B.17}$$

$$\boldsymbol{p}_{n+1} - \boldsymbol{p}_n \propto \frac{\partial \mathscr{U}\left(\boldsymbol{q}_{n+\alpha}\right)}{\partial \boldsymbol{q}_{n+\alpha}} = \nabla \mathscr{U}\left(\boldsymbol{q}_{n+\alpha}\right) \tag{B.18}$$

Therefore, introducing aribitrary functions $\kappa_\ell : T^*Q \to \mathbb{R}$ $(\ell = 1, 2)$, the family of algorithms which conserve the angular momentum within time step $[t_n, t_{n+1}]$ exactly is given as follows (note $\Delta t := t_{n+1} - t_n > 0$ denotes the time step size):

Algorithm 33

Family of single-step angular momentum conserving algorithms

Given $\boldsymbol{q}_n$ *and* $\boldsymbol{p}_n$*, find* $\boldsymbol{q}_{n+1}$ *and* $\boldsymbol{p}_{n+1}$ *from*

$$\boldsymbol{q}_{n+1} = \boldsymbol{q}_n + \Delta t \kappa_1 \boldsymbol{M}^{-1} \boldsymbol{p}_{n+(1-\alpha)} \tag{B.19a}$$

$$\boldsymbol{p}_{n+1} = \boldsymbol{p}_n - \Delta t \kappa_2 \nabla \mathscr{U}\left(\boldsymbol{q}_{n+\alpha}\right) \tag{B.19b}$$

for $\alpha \in [0,1] \subset \mathbb{R}$ *and* $\kappa_\ell : T^*Q \to \mathbb{R}$ $(\ell = 1,2)$.

Remark 34 (Algorithm 33)

1. *Consistency condition: The arbitrary functions* κ_ℓ *need to satisfy* $\kappa_\ell = 1 + \mathcal{O}(\Delta t)$ *for sufficiently small time step size, i.e.,* $\lim_{\Delta t \to 0} \kappa_\ell = 1$.

2. *When* $\kappa_1 = \kappa_2 = 1$ *and* $\alpha \in [0,1]$, *the algorithms are a family of single-step symplectic schemes:*

$$\begin{aligned} \boldsymbol{q}_{n+1} &= \boldsymbol{q}_n + \Delta t \boldsymbol{M}^{-1} \boldsymbol{p}_{n+(1-\alpha)} && \text{(B.20a)} \\ \boldsymbol{p}_{n+1} &= \boldsymbol{p}_n - \Delta t \nabla \mathscr{U}\left(\boldsymbol{q}_{n+\alpha}\right) && \text{(B.20b)} \end{aligned}$$

Equations (B.20) *are second-order time accurate and unconditionally stable only when* $\alpha = 1/2$ *(Mid-point rule); otherwise, they are first-order time accurate and conditionally stable of critical time step* $\Delta t_{\text{crit}} = 1/\left(\mid \alpha - \frac{1}{2} \mid \Omega_{\max}\right)$ *where* $\Omega_{\max}$ *is the maximum frequency of the linearized system.*

From equation (B.1), finite increment of the kinetic energy within a time step $[t_n, t_{n+1}]$ can be written as

$$\begin{aligned} \mathscr{T}\left(\boldsymbol{p}_{n+1}\right) - \mathscr{T}\left(\boldsymbol{p}_n\right) &= \left[\boldsymbol{p}_{n+1} - \boldsymbol{p}_n\right] \cdot \boldsymbol{M}^{-1} \left[\frac{\boldsymbol{p}_{n+1} + \boldsymbol{p}_n}{2}\right] \\ &= \Delta \boldsymbol{p} \cdot \boldsymbol{M}^{-1} \boldsymbol{p}_{n+1/2} && \text{(B.21)} \end{aligned}$$

where $\Delta \boldsymbol{p} := \boldsymbol{p}_{n+1} - \boldsymbol{p}_n$. Substituting $\boldsymbol{p}_{n+1/2} = \boldsymbol{p}_{n+(1-\alpha)} + \left(\alpha - \frac{1}{2}\right)\left(\boldsymbol{p}_{n+1} - \boldsymbol{p}_n\right)$, the right-hand side of the above equation yields

$$\text{RHS} = \Delta \boldsymbol{p} \cdot \boldsymbol{M}^{-1} \boldsymbol{p}_{n+(1-\alpha)} + \left(\alpha - \tfrac{1}{2}\right) \Delta \boldsymbol{p} \cdot \boldsymbol{M}^{-1} \Delta \boldsymbol{p} \qquad \text{(B.22)}$$

In view of Algorithm 33, we get

$$\begin{aligned} \mathscr{T}\left(\boldsymbol{p}_{n+1}\right) - \mathscr{T}\left(\boldsymbol{p}_n\right) &= \left(\alpha - \tfrac{1}{2}\right) \Delta \boldsymbol{p} \cdot \boldsymbol{M}^{-1} \Delta \boldsymbol{p} - \tfrac{\kappa_2}{\kappa_1} \nabla \mathscr{U}\left(\boldsymbol{q}_{n+\alpha}\right) \cdot \left[\boldsymbol{q}_{n+1} - \boldsymbol{q}_n\right] \\ &= \kappa_2^2 \Delta t^2 \left(\alpha - \tfrac{1}{2}\right) \nabla \mathscr{U}\left(\boldsymbol{q}_{n+\alpha}\right) \cdot \boldsymbol{M}^{-1} \nabla \mathscr{U}\left(\boldsymbol{q}_{n+\alpha}\right) \\ &\quad - \tfrac{\kappa_2}{\kappa_1} \nabla \mathscr{U}\left(\boldsymbol{q}_{n+\alpha}\right) \cdot \left[\boldsymbol{q}_{n+1} - \boldsymbol{q}_n\right] && \text{(B.23)} \end{aligned}$$

Enforcing the exact energy conserving in $[t_n, t_{n+1}]$; namely,

$$\mathscr{T}\left(\boldsymbol{p}_{n+1}\right) - \mathscr{T}\left(\boldsymbol{p}_n\right) = -\left[\mathscr{U}\left(\boldsymbol{q}_{n+1}\right) - \mathscr{U}\left(\boldsymbol{q}_n\right)\right] \tag{B.24}$$

the following relation is readily obtained:

$$\boxed{\begin{aligned}\mathscr{U}\left(\boldsymbol{q}_{n+1}\right) - \mathscr{U}\left(\boldsymbol{q}_n\right) =& \tfrac{\kappa_2}{\kappa_1}\nabla\mathscr{U}\left(\boldsymbol{q}_{n+\alpha}\right)\cdot\left[\boldsymbol{q}_{n+1} - \boldsymbol{q}_n\right] \\ &+ \left(\tfrac{1}{2} - \alpha\right)|||\,\nabla\mathscr{U}\left(\boldsymbol{q}_{n+\alpha}\right)\,|||^2\,\kappa_2^2\Delta t^2\end{aligned}} \tag{B.25}$$

where $|||\,\nabla\mathscr{U}\left(\boldsymbol{q}_{n+\alpha}\right)\,|||:=\sqrt{\nabla\mathscr{U}\left(\boldsymbol{q}_{n+\alpha}\right)\cdot\boldsymbol{M}^{-1}\nabla\mathscr{U}\left(\boldsymbol{q}_{n+\alpha}\right)}$. Hence, the family of single-step exact energy-momentum conserving algorithms may be summarized as follows:

Algorithm 34

Family of single-step energy-momentum conserving algorithms

Given $\boldsymbol{q}_n$ *and* $\boldsymbol{p}_n$*, find* $\boldsymbol{q}_{n+1}$ *and* $\boldsymbol{p}_{n+1}$ *from*

$$\boldsymbol{q}_{n+1} = \boldsymbol{q}_n + \Delta t\kappa_1\boldsymbol{M}^{-1}\boldsymbol{p}_{n+(1-\alpha)} \tag{B.26a}$$

$$\boldsymbol{p}_{n+1} = \boldsymbol{p}_n - \Delta t\kappa_2\nabla\mathscr{U}\left(\boldsymbol{q}_{n+\alpha}\right) \tag{B.26b}$$

with

$$\begin{aligned}\mathscr{U}\left(\boldsymbol{q}_{n+1}\right) - \mathscr{U}\left(\boldsymbol{q}_n\right) =& \tfrac{\kappa_2}{\kappa_1}\nabla\mathscr{U}\left(\boldsymbol{q}_{n+\alpha}\right)\cdot\left[\boldsymbol{q}_{n+1} - \boldsymbol{q}_n\right] \\ &+ \left(\tfrac{1}{2} - \alpha\right)|||\,\nabla\mathscr{U}\left(\boldsymbol{q}_{n+\alpha}\right)\,|||^2\,\kappa_2^2\Delta t^2\end{aligned} \tag{B.26c}$$

for $\alpha \in [0,1] \subset \mathbb{R}$ *and* $\kappa_\ell : T^*Q \to \mathbb{R}$ $(\ell = 1, 2)$.

Remark 35 (Algorithm 34)

1. *Suppose equation* (B.26) *is solvable; then, Algorithm 34 is unconditionally stable in the sense of exact energy conserving within time step. Remember the total energy defines the Lyapunov function of the dynamical system.*

2. *Projection methods: Setting* $\alpha = 1/2$ *and* $\kappa_1 = 1$*, Algorithm 34 is reduced to the following second-order time accurate algorithms:*

$$\boldsymbol{q}_{n+1} = \boldsymbol{q}_n + \Delta t\boldsymbol{M}^{-1}\boldsymbol{p}_{n+1/2} \tag{B.27a}$$

$$\boldsymbol{p}_{n+1} = \boldsymbol{p}_n - \Delta t\kappa_2\nabla\mathscr{U}\left(\boldsymbol{q}_{n+1/2}\right) \tag{B.27b}$$

and solve κ_2 *from*

$$\mathscr{U}\left(\boldsymbol{q}_{n+1}\right) - \mathscr{U}\left(\boldsymbol{q}_n\right) = \kappa_2 \nabla \mathscr{U}\left(\boldsymbol{q}_{n+1/2}\right) \cdot \left[\boldsymbol{q}_{n+1} - \boldsymbol{q}_n\right] \tag{B.27c}$$

For linear Hamiltonian systems, we have $\kappa_2 = 1$*, and the energy conserving property of the Crank-Nicolson scheme is recovered.*

3. *Collocation schemes: Setting* $\kappa_1 = \kappa_2 = 1$*, Algorithm 34 is reduced to*

$$\begin{aligned} \boldsymbol{q}_{n+1} &= \boldsymbol{q}_n + \Delta t \boldsymbol{M}^{-1} \boldsymbol{p}_{n+(1-\alpha)} && \text{(B.28a)} \\ \boldsymbol{p}_{n+1} &= \boldsymbol{p}_n - \Delta t \nabla \mathscr{U}\left(\boldsymbol{q}_{n+\alpha}\right) && \text{(B.28b)} \end{aligned}$$

with

$$\begin{aligned} \mathscr{U}\left(\boldsymbol{q}_{n+1}\right) - \mathscr{U}\left(\boldsymbol{q}_n\right) = & \nabla \mathscr{U}\left(\boldsymbol{q}_{n+\alpha}\right) \cdot \left[\boldsymbol{q}_{n+1} - \boldsymbol{q}_n\right] \\ & + \left(\tfrac{1}{2} - \alpha\right) |||\, \nabla \mathscr{U}\left(\boldsymbol{q}_{n+\alpha}\right) |||^2 \, \Delta t^2 \end{aligned} \tag{B.28c}$$

Note there exists at least one $\alpha \in [0, 1]$ *which satisfies equation* (B.28c)*. For linear Hamiltonian systems, we have* $\alpha = 1/2$*, and the energy conserving property of Crank-Nicolson scheme is recovered.*

4. *For linear Hamiltonian systems* $\alpha = 1/2$ *with* $\kappa_1 = \kappa_2 = 1$*, Algorithm 34 becomes the Mid-point rule which is second-order time accurate, unconditionally stable, symplectic, and energy-momentum conserving. It is important to note that the properties of symplecticness and energy conserving within time step co-exist for this case.*

B.2 Energy-Momentum Conserving Algorithm for Nonlinear Elastodynamics

B.2.1 Simo-Tarnow Framework

In this section, we summarize the exact energy-momentum conserving algorithm designed for a compressible hyperelastic material model, described in Simo and

Tarnow [55]. Although this framework has been designed for a compressible general hyperelastic material model, the algorithm has been implemented only for St. Venant-Kirchhoff material model because of its disadvantages with being numerically robust. However, the numerical difficulties have been addressed by Laursen and Meng [67], and the framework can be used for compressible general hyperelastic material model with the numerical implementations proposed by Laursen and Meng [67].

Consider a partition $\mathbb{I} = [t_0, T] = \bigcup_{n=0}^{k-1} [t_n, t_{n+1}]$ with $t_k = T$, and the focus is on a typical time interval $[t_n, t_{n+1}] \subset \mathbb{I}$. Define the convex combinations of configurations and the material veocity fields by

$$\boldsymbol{\varphi}_{n+\alpha} := \alpha \boldsymbol{\varphi}_{n+1} + (1-\alpha)\, \boldsymbol{\varphi}_n \in Q \tag{B.29}$$

$$\boldsymbol{V}_{n+\alpha} := \alpha \boldsymbol{V}_{n+1} + (1-\alpha)\, \boldsymbol{V}_n \tag{B.30}$$

for $\alpha \in [0,1]$. The smooth manifold of admissible configurations, Q, may be defined by

$$Q := \left\{ \boldsymbol{\varphi} : \Omega_0 \to \mathbb{R}^{\mathrm{n_{dim}}} \mid J := \det[\boldsymbol{F}] > 0 \text{ and } \boldsymbol{\varphi} \mid_{\Gamma_0^{\boldsymbol{\varphi}}} = \bar{\boldsymbol{\varphi}} \right\} \tag{B.31}$$

From the weak form of the balance of linear momentum, i.e.,

$$\int_{\Omega_0} \rho_0 \dot{\boldsymbol{V}} \cdot \boldsymbol{\eta} d\Omega_0 + \int_{\Omega_0} \boldsymbol{P} \cdot\cdot \nabla_0 \boldsymbol{\eta} d\Omega_0 = \int_{\Omega_0} \rho_0 \boldsymbol{b} \cdot \boldsymbol{\eta} d\Omega_0 + \int_{\Gamma_0^\sigma} \bar{\boldsymbol{t}}_0 \cdot \boldsymbol{\eta} d\Gamma_0^\sigma \tag{B.32}$$

Consider the following algorithmic approximation:

$$\begin{aligned} &\int_{\Omega_0} \rho_0 \frac{\boldsymbol{V}_{n+1} - \boldsymbol{V}_n}{\Delta t} \cdot \boldsymbol{\eta} d\Omega_0 + \int_{\Omega_0} \tilde{\boldsymbol{S}} \cdot \boldsymbol{F}^T \left(\boldsymbol{\varphi}_{n+\alpha} \right) \cdot\cdot \nabla_0 \boldsymbol{\eta} d\Omega_0 \\ &= \int_{\Omega_0} \rho_0 \boldsymbol{b}_{n+\alpha} \cdot \boldsymbol{\eta} d\Omega_0 + \int_{\Gamma_0^\sigma} \bar{\boldsymbol{t}}^0_{n+\alpha} \cdot \boldsymbol{\eta} d\Gamma_0^\sigma, \quad \forall \boldsymbol{\eta} \in \mathcal{V} \end{aligned} \tag{B.33a}$$

with the approximated local relation

$$\frac{\boldsymbol{\varphi}_{n+1} - \boldsymbol{\varphi}_n}{\Delta t} = \boldsymbol{V}_{n+\alpha} \tag{B.33b}$$

and we assume initial conditions are given; that is,

$$\int_{\Omega_0} \boldsymbol{\varphi}(\boldsymbol{X}, t_0) \cdot \boldsymbol{\eta} d\Omega_0 = \int_{\Omega_0} \boldsymbol{\varphi}_0 \cdot \boldsymbol{\eta} d\Omega_0, \quad \boldsymbol{X} \in \Omega_0 \tag{B.33c}$$

$$\int_{\Omega_0} \boldsymbol{V}\left(\boldsymbol{X}, t_0\right) \cdot \boldsymbol{\eta} d\Omega_0 = \int_{\Omega_0} \boldsymbol{V}_0 \cdot \boldsymbol{\eta} d\Omega_0, \quad \boldsymbol{X} \in \Omega_0 \tag{B.33d}$$

Recall that the space of test functions may be defined as

$$\mathcal{V} := \left\{ \boldsymbol{\eta} : \Omega_0 \to \mathbb{R}^{\mathrm{n_{dim}}} \mid \boldsymbol{\eta} \mid_{\Gamma_0^{\boldsymbol{\varphi}}} = \mathbf{0} \right\} \tag{B.34}$$

We will achieve the exact energy-momentum conserving algorithms within a time step $[t_n, t_{n+1}]$ under the framework of equation (B.33) in the sense of

$$\boldsymbol{L}_n = \boldsymbol{L}_{n+1}, \quad \boldsymbol{J}_n = \boldsymbol{J}_{n+1} \quad (n = 0, 1, \ldots, k-1) \tag{B.35a}$$

where $\boldsymbol{L}$ and $\boldsymbol{J}$ denote the linear and angular momenta, respectively, and

$$\mathscr{E}_n \geq \mathscr{E}_{n+1} \text{ and } \mathscr{E}_n \equiv \mathscr{E}_{n+1} \Leftrightarrow \mathscr{D}^{\mathrm{int}} \equiv 0 \quad (n = 0, 1, \ldots, k-1) \tag{B.35b}$$

where $\mathscr{E}$ and $\mathscr{D}^{\mathrm{int}}$ denotes the total energy and the internal dissipation function of the system, respectively. We need to find the conditions for the parameter $\alpha \in [0, 1]$ which satisfy the conditions given by equation (B.35). Note $\boldsymbol{L}_n := \boldsymbol{L}\left(\boldsymbol{V}_n\right)$ and $\boldsymbol{L}_{n+1} := \boldsymbol{L}\left(\boldsymbol{V}_{n+1}\right)$; $\boldsymbol{J}_n := \boldsymbol{J}\left(\boldsymbol{\varphi}_n, \boldsymbol{V}_n\right)$ and $\boldsymbol{J}_{n+1} := \boldsymbol{J}\left(\boldsymbol{\varphi}_{n+1}, \boldsymbol{V}_{n+1}\right)$; $\mathscr{E}_n := \mathscr{E}\left(\boldsymbol{\varphi}_n, \boldsymbol{V}_n\right)$ and $\mathscr{E}_{n+1} := \mathscr{E}\left(\boldsymbol{\varphi}_{n+1}, \boldsymbol{V}_{n+1}\right)$.

Theorem 6

Time Discrete momentum conservation within a time step

Suppose the algorithmic 2nd Piola-Kirchhoff stress $\tilde{\boldsymbol{S}}$ is arbitrary and symmetric ($\tilde{\boldsymbol{S}}^T = \tilde{\boldsymbol{S}}$). Then, in the absence of the rsultant external loading $\boldsymbol{F}^{\mathrm{ext}}_{n+\alpha} = \mathbf{0}$ and $\boldsymbol{T}^{\mathrm{ext}}_{n+\alpha} = \mathbf{0}$, the discrete linear momentum and the discrete angular momentum are conserved in $[t_n, t_{n+1}$ for any $\alpha \in [0, 1]$ and for $\alpha = 1/2$, respectively ($\boldsymbol{T}^{\mathrm{ext}}_{n+\alpha} := \boldsymbol{\varphi}_{n+\alpha} \times \boldsymbol{F}^{\mathrm{ext}}_{n+\alpha}$ denotes the resultant momentum due to the external loading). Therefore, for the satisfaction of the conservation of time discrete linear and angular momenta, we must have $\alpha = 1/2$:

$$\boxed{\boldsymbol{F}^{\mathrm{ext}}_{n+\alpha} = \mathbf{0},\ \boldsymbol{T}^{\mathrm{ext}}_{n+\alpha} = \mathbf{0},\ \textit{and}\ \alpha = \tfrac{1}{2} \Rightarrow \boldsymbol{L}_n = \boldsymbol{L}_{n+1}\ \textit{and}\ \boldsymbol{J}_n = \boldsymbol{J}_{n+1}} \tag{B.36a}$$

Proof 1 (Proposition 6)

1. *Time discrete linear momentum conservation: Setting* $\boldsymbol{\eta} = \boldsymbol{\xi}\,(=\text{const.}) \in \Omega_0$ *such that* $\nabla_0 \boldsymbol{\eta} = \nabla_0 \boldsymbol{\xi} = \boldsymbol{0}$ *in equation* (B.33a) *yields*

$$\int_{\Omega_0} \rho_0 \frac{\boldsymbol{V}_{n+1} - \boldsymbol{V}_n}{\Delta t} \cdot \boldsymbol{\xi} d\Omega_0 = \underbrace{\int_{\Omega_0} \rho_0 \boldsymbol{b}_{n+\alpha} \cdot \boldsymbol{\xi} d\Omega_0 + \int_{\Gamma_0^\sigma} \bar{\boldsymbol{t}}^0_{n+\alpha} \cdot \boldsymbol{\xi} d\Gamma_0^\sigma}_{\boldsymbol{F}^{\text{ext}}_{n+\alpha} \cdot \boldsymbol{\xi}} \tag{B.37}$$

With the definition of the linear momentum; namely, $\boldsymbol{L} := \int_{\Omega_0} \rho_0 \boldsymbol{V} d\Omega_0$, *we get*

$$\frac{\boldsymbol{L}_{n+1} - \boldsymbol{L}_n}{\Delta t} \cdot \boldsymbol{\xi} = \boldsymbol{F}^{\text{ext}}_{n+\alpha} \cdot \boldsymbol{\xi} \tag{B.38}$$

Hence, the discrete linear momentum is conserved in $[t_n, t_{n+1}]$ *for any* $\alpha \in [0,1]$ *in the absence of the resultant external loading.*

2. *Time discrete angular momentum conservation: Setting* $\boldsymbol{\eta} = \boldsymbol{\xi} \times \boldsymbol{\varphi}_{n+\alpha}$, *where* $\boldsymbol{\xi}\,(=\text{const.}) \in \mathbb{R}^3$, *such that* $\nabla_0 \boldsymbol{\eta} = \hat{\boldsymbol{\eta}} \boldsymbol{F}\left(\boldsymbol{\varphi}_{n+\alpha}\right)$, *equation* (B.33a) *yields*

$$\begin{aligned}
&\int_{\Omega_0} \rho_0 \frac{\boldsymbol{V}_{n+1} - \boldsymbol{V}_n}{\Delta t} \cdot \left[\boldsymbol{\xi} \times \boldsymbol{\varphi}_{n+\alpha}\right] d\Omega_0 \\
&+ \int_{\Omega_0} \tilde{\boldsymbol{S}} \cdot \boldsymbol{F}^T\left(\boldsymbol{\varphi}_{n+\alpha}\right) \cdot\cdot \left[\hat{\boldsymbol{\eta}} \boldsymbol{F}\left(\boldsymbol{\varphi}_{n+\alpha}\right)\right] d\Omega_0 \\
&= \int_{\Omega_0} \rho_0 \boldsymbol{b}_{n+\alpha} \cdot \left[\boldsymbol{\xi} \times \boldsymbol{\varphi}_{n+\alpha}\right] d\Omega_0 + \int_{\Gamma_0^\sigma} \bar{\boldsymbol{t}}^0_{n+\alpha} \cdot \left[\boldsymbol{\xi} \times \boldsymbol{\varphi}_{n+\alpha}\right] d\Gamma_0^\sigma \\
\Rightarrow \quad &\int_{\Omega_0} \left[\boldsymbol{\varphi}_{n+\alpha} \times \rho_0 \left(\boldsymbol{V}_{n+1} - \boldsymbol{V}_n\right)\right] \cdot \boldsymbol{\xi} d\Omega_0 \\
&= \Delta t \underbrace{\left(\int_{\Omega_0} \left[\boldsymbol{\varphi}_{n+\alpha} \times \rho_0 \boldsymbol{b}_{n+\alpha}\right] \cdot \boldsymbol{\eta} d\Omega_0 + \int_{\Gamma_0^\sigma} \left[\boldsymbol{\varphi}_{n+\alpha} \times \bar{\boldsymbol{t}}^0_{n+\alpha}\right] \cdot \boldsymbol{\eta} d\Gamma_0^\sigma \right)}_{\boldsymbol{T}^{\text{ext}}_{n+\alpha} \cdot \boldsymbol{\xi}} \\
&- \Delta t \underbrace{\int_{\Omega_0} \left[\boldsymbol{F}^T\left(\boldsymbol{\varphi}_{n+\alpha}\right) \cdot \tilde{\boldsymbol{S}} \cdot \boldsymbol{F}\left(\boldsymbol{\varphi}_{n+\alpha}\right)\right] \cdot\cdot \hat{\boldsymbol{\xi}} d\Omega_0}_{0} \\
&= \Delta t \boldsymbol{\xi} \cdot \boldsymbol{T}^{\text{ext}}_{n+\alpha}
\end{aligned} \tag{B.39}$$

Using the definition of the angular momentum; namely, $\boldsymbol{J} := \int_{\Omega_0} \boldsymbol{\varphi} \times \rho_0 \boldsymbol{V} d\Omega_0$,

$$\begin{aligned}
\boldsymbol{\xi} \cdot (\boldsymbol{J}_{n+1} - \boldsymbol{J}_n) &= \boldsymbol{\xi} \cdot \int_{\Omega_0} \left(\boldsymbol{\varphi}_{n+1} \times \rho_0 \boldsymbol{V}_{n+1} - \boldsymbol{\varphi}_n \times \rho_0 \boldsymbol{V}_n \right) d\Omega_0 \\
&= \boldsymbol{\xi} \cdot \int_{\Omega_0} \boldsymbol{\varphi}_{n+\alpha} \times \rho_0 \left(\boldsymbol{V}_{n+1} - \boldsymbol{V}_n \right) d\Omega_0 \\
&\quad + \boldsymbol{\xi} \cdot \int_{\Omega_0} \left(\boldsymbol{\varphi}_{n+1} - \boldsymbol{\varphi}_n \right) \times \rho_0 \boldsymbol{V}_{n+\alpha} d\Omega_0 \\
&= \boldsymbol{\xi} \cdot \int_{\Omega_0} \boldsymbol{\varphi}_{n+\alpha} \times \rho_0 \left(\boldsymbol{V}_{n+1} - \boldsymbol{V}_n \right) d\Omega_0 \\
&\quad + \boldsymbol{\xi} \cdot 2\Delta t \int_{\Omega_0} \rho_0 \boldsymbol{V}_{n+\alpha} \times \boldsymbol{V}_{n+1/2} d\Omega_0 \qquad \text{(B.40)}
\end{aligned}$$

Notice equation (B.33b) *has been used in the last step above. Substituting equation* (B.39) *into the first term of equation* (B.57) *yields*

$$\boldsymbol{\xi} \cdot (\boldsymbol{J}_{n+1} - \boldsymbol{J}_n) = \Delta t \boldsymbol{\xi} \cdot \boldsymbol{T}^{\text{ext}}_{n+\alpha} + 2\Delta t \boldsymbol{\xi} \cdot \int_{\Omega_0} \rho_0 \boldsymbol{V}_{n+\alpha} \times \boldsymbol{V}_{n+1/2} d\Omega_0 \qquad \text{(B.41)}$$

When $\alpha = 1/2$, *equation* (B.42) *becomes*

$$\frac{\boldsymbol{J}_{n+1} - \boldsymbol{J}_n}{\Delta t} \cdot \boldsymbol{\xi} = \boldsymbol{T}^{\text{ext}}_{n+1/2} \cdot \boldsymbol{\xi} \qquad \text{(B.42)}$$

Hence, the discrete angular momentum is conserved in $[t_n, t_{n+1}]$ *only if* $\alpha = 1/2$ *in the absence of the resultant external momentum (torque).*

Therefore, the exact momentum conserving algorithm within a time step $[t_n, t_{n+1}]$ may be written as

Algorithm 35

Momentum conserving algorithm

Suppose the algorithmic 2nd Piola-Kirchhoff stress $\tilde{\boldsymbol{S}}$ *is arbitrary and symmetric* ($\tilde{\boldsymbol{S}}^T = \tilde{\boldsymbol{S}}$); *then, the following algorithm conserves the linear and angular momenta*

within a time step $[t_n, t_{n+1}]$*:*

$$\int_{\Omega_0} \rho_0 \frac{\boldsymbol{V}_{n+1} - \boldsymbol{V}_n}{\Delta t} \cdot \boldsymbol{\eta} d\Omega_0 + \int_{\Omega_0} \tilde{\boldsymbol{S}} \cdot \boldsymbol{F}^T \left(\boldsymbol{\varphi}_{n+1/2}\right) \cdot\cdot \nabla_0 \boldsymbol{\eta} d\Omega_0$$
$$= \int_{\Omega_0} \rho_0 \boldsymbol{b}_{n+1/2} \cdot \boldsymbol{\eta} d\Omega_0 + \int_{\Gamma_0^\sigma} \bar{\boldsymbol{t}}^0_{n+1/2} \cdot \boldsymbol{\eta} d\Gamma_0^\sigma \tag{B.43a}$$
$$\int_{\Omega_0} \frac{\boldsymbol{\varphi}_{n+1} - \boldsymbol{\varphi}_n}{\Delta t} d\Omega_0 = \int_{\Omega_0} \boldsymbol{V}_{n+1/2} d\Omega_0 \tag{B.43b}$$

$\forall \boldsymbol{\eta} \in \mathcal{V}$.

Remark 36 (Algorithm 35)

1. *It is important to note that the discrete linear and angular momenta are conserved with an arbitrary and symmetric algorithmic 2nd Piola-Kirchhoff stress* $\tilde{\boldsymbol{S}}$*.*

2. *Algorithm 35 satisfies the algorithmic form of the theorem of expended power within a time step* $[t_n, t_{n+1}]$*: Setting* $\boldsymbol{\eta} = \boldsymbol{V}_{n+1/2}$ *in equation* (B.43a) *yields*

$$\underbrace{\int_{\Omega_0} \rho_0 \left(\boldsymbol{V}_{n+1} - \boldsymbol{V}_n\right) \cdot \boldsymbol{V}_{n+1/2} d\Omega_0}_{\mathscr{T}_{n+1} - \mathscr{T}_n}$$
$$+ \int_{\Omega_0} \tilde{\boldsymbol{S}} \cdot \underbrace{\boldsymbol{F}^T \left(\boldsymbol{\varphi}_{n+1/2}\right) \cdot\cdot \nabla_0 \left(\Delta t \boldsymbol{V}_{n+1/2}\right)}_{\frac{1}{2}\left[\boldsymbol{C}\left(\boldsymbol{\varphi}_{n+1}\right) - \boldsymbol{C}\left(\boldsymbol{\varphi}_n\right)\right]} d\Omega_0$$
$$= \Delta t \int_{\Omega_0} \rho_0 \boldsymbol{b}_{n+1/2} \cdot \boldsymbol{V}_{n+1/2} d\Omega_0 + \Delta t \int_{\Gamma_0^\sigma} \bar{\boldsymbol{t}}^0_{n+1/2} \cdot \boldsymbol{V}_{n+1/2} d\Gamma_0^\sigma \tag{B.44}$$

   where $\mathscr{T}$ *and* $\boldsymbol{C}$ *denotes the kinetic energy of the system and the right Cauchy-Green strain tensor, respectively, and the right-hand side implies the change of the potential energy loading within a time step.*

To study the criteria of the discrete energy balance within a time step $[t_n, t_{n+1}]$, consider the following theorem:

Theorem 7 (Energy balance)

In a mechanical system, the following inequality is true:

$$\boxed{\dot{\mathscr{E}} - \mathscr{P}^{\text{ext}} = -\mathscr{D}^{\text{int}} \leq 0, \quad \forall t \in \mathbb{I}} \tag{B.45a}$$

where $\mathscr{D}^{\text{int}} \geq 0$ *and*

$$\mathscr{P}^{\text{ext}} = \int_{\Omega_0} \rho_0 \boldsymbol{b}\left(\boldsymbol{X}, t\right) \cdot \boldsymbol{V}\left(\boldsymbol{X}, t\right) d\Omega_0 + \int_{\Gamma_0^\sigma} \bar{\boldsymbol{t}}_0\left(\boldsymbol{X}, t\right) \cdot \boldsymbol{V}\left(\boldsymbol{X}, t\right) d\Gamma_0^\sigma \tag{B.45b}$$

denote the internal dissipation function and the expended power of the external loading of the system, respectively. The total energy of the system $\mathscr{E}$ *is defined by the sum of the kinetic energy* $\mathscr{T}$ *and the internal energy* $\mathscr{U}$*:*

$$\mathscr{E} := \mathscr{T} + \mathscr{U} = \int_{\Omega_0} \tfrac{1}{2}\rho_0 \boldsymbol{V}\left(\boldsymbol{X}, t\right) \cdot \boldsymbol{V}\left(\boldsymbol{X}, t\right) d\Omega_0 + \textstyle\int_{\Omega_0} W d\Omega_0 \tag{B.45c}$$

where W *denotes the stored energy density function defined per unit reference volume.*

Proof 2 (Theorem 7)

Consider the first law of thermodynamics in a mechanical system ($\mathscr{P}^{\text{heat}} = 0$*), i.e.,*

$$\dot{\mathscr{E}} = \mathscr{P}^{\text{ext}} \tag{B.46}$$

With equation (B.45c)*, the left-hand side of above equation can be also written as*

$$\dot{\mathscr{E}} = \dot{\mathscr{K}} + \dot{\mathscr{U}} = \int_{\Omega_0} \left[\rho_0 \boldsymbol{V}\left(\boldsymbol{X}, t\right) \cdot \frac{\partial \boldsymbol{V}\left(\boldsymbol{X}, t\right)}{\partial t} + \frac{\partial W}{\partial t}\right] d\Omega_0 \tag{B.47}$$

According to the Clausius-Planck inequality, the internal dissipation function in the local form is given by

$$\mathcal{D}^{\text{int}} := \boldsymbol{P} \cdot\cdot \dot{\boldsymbol{F}}^T - \frac{\partial W}{\partial t} \geq 0 \tag{B.48}$$

in the mechanical system ($s_0 = 0$*). In the continuum system, the internal dissipation function may be given by*

$$\begin{aligned} \mathscr{D}^{\text{int}} &:= \int_{\Omega_0} \boldsymbol{P} \cdot\cdot \dot{\boldsymbol{F}}^T d\Omega_0 - \int_{\Omega_0} \frac{\partial W}{\partial t} d\Omega_0 \\ &= \mathscr{P}^{\text{int}} - \dot{\mathscr{U}} \geq 0 \end{aligned} \tag{B.49}$$

where $\mathscr{P}^{\text{int}} := \int_{\Omega_0} \boldsymbol{P} \cdot\cdot \dot{\boldsymbol{F}}^T d\Omega_0$ *is called the total stress power of the system. Recalling the following relation which is true in the mechanical system:*

$$\dot{\mathscr{U}} = \int_{\Omega_0} \frac{\partial W}{\partial t} d\Omega_0 = \int_{\Omega_0} \boldsymbol{P} \cdot\cdot \dot{\boldsymbol{F}}^T d\Omega_0 \tag{B.50}$$

inequality (B.45a) *can be readily derived from equations* (B.46), (B.47) *and* (B.49).

In view of equation (B.44), the discrete form of theorem 7 in the absense of the external loading (therefore, $\mathscr{P}^{\text{ext}} = 0$) may be written as

$$\mathscr{U}_{n+1} - \mathscr{U}_n = \int_{\Omega_0} \tilde{\boldsymbol{S}} : \tfrac{1}{2} \left[\boldsymbol{C}\left(\boldsymbol{\varphi}_{n+1}\right) - \boldsymbol{C}\left(\boldsymbol{\varphi}_n\right) \right] d\Omega_0 - \Delta\mathscr{D}^{\text{int}} \;\text{ with }\; \Delta\mathscr{D}^{\text{int}} \geq 0 \tag{B.51}$$

For the exact energy conservation within a time step ($\Delta\mathscr{D}^{\text{int}} \equiv 0$), the following relation must hold:

$$\boxed{\mathscr{U}_{n+1} - \mathscr{U}_n = \int_{\Omega_0} \tilde{\boldsymbol{S}} : \tfrac{1}{2} \left[\boldsymbol{C}\left(\boldsymbol{\varphi}_{n+1}\right) - \boldsymbol{C}\left(\boldsymbol{\varphi}_n\right) \right] d\Omega_0} \tag{B.52}$$

To observe the expression of the algorithmic 2nd Piola-Kirchhoff stress $\tilde{\boldsymbol{S}}$ for the exact energy-momentum conserving algorithm for smooth nonlinear elasto-dynamics with the restriction of equation (B.52), define the *algorithmic* right Cauchy-Green strain tensor as

$$\boldsymbol{C}_{n+\beta} := \beta \boldsymbol{C}\left(\boldsymbol{\varphi}_n\right) + \left[1-\beta\right] \boldsymbol{C}\left(\boldsymbol{\varphi}_{n+1}\right) \tag{B.53}$$

for $\beta \in [0,1]$. Note $\boldsymbol{C}\left(\boldsymbol{\varphi}_n\right) \equiv \boldsymbol{C}_n$ and $\boldsymbol{C}\left(\boldsymbol{\varphi}_{n+1}\right) \equiv \boldsymbol{C}_{n+1}$; however, $\boldsymbol{C}(\boldsymbol{\varphi}_{n+\beta}) \neq \boldsymbol{C}_{n+\beta}$ for $\beta \in (0,1)$ in general. It is very important to use $\boldsymbol{C}_{n+\beta}$ intead of $\boldsymbol{C}(\boldsymbol{\varphi}_{n+\beta})$ to avoid non-physical couplings in the calculation which may causes numerical instabilities; see Gonzalez and Simo [68]. Note the algorithmic right Cauchy-Green strain tensor remains symmetric and positive definite for all $\beta \in [0,1]$. By the definition, the algorithmic Lagrange-Green strain tensor may be correspondingly written as

$$\boldsymbol{E}_{n+\beta} := \tfrac{1}{2} \left(\boldsymbol{C}_{n+\beta} - \boldsymbol{I} \right) \tag{B.54}$$

For the exact energy conservation within a time step $[t_n, t_{n+1}]$, the mean value theorem must be used; namely, there exist some $\beta_0 \in (0,1)$ such that

$$\begin{aligned} \hat{W}\left(\boldsymbol{C}_{n+1}\right) - \hat{W}\left(\boldsymbol{C}_n\right) &= \frac{\partial \hat{W}\left(\boldsymbol{C}_{n+\beta_0}\right)}{\partial \boldsymbol{C}_{n+\beta_0}} : \left[\boldsymbol{C}_{n+1} - \boldsymbol{C}_n\right] \\ &= \nabla \hat{W}\left(\boldsymbol{C}_{n+\beta_0}\right) : \left[\boldsymbol{C}_{n+1} - \boldsymbol{C}_n\right] \end{aligned} \tag{B.55}$$

Comparison with with the absense of external loading and $\Delta \mathscr{D}^{\text{int}} \equiv 0$, i.e.,

$$\int_{\Omega_0} \left[\hat{W}\left(\boldsymbol{C}_{n+1}\right) - \hat{W}\left(\boldsymbol{C}_n\right)\right] d\Omega_0 = \tfrac{1}{2} \int_{\Omega_0} \tilde{\boldsymbol{S}} : \left[\boldsymbol{C}_{n+1} - \boldsymbol{C}_n\right] d\Omega_0 \tag{B.56}$$

yields

$$\boxed{\tilde{\boldsymbol{S}} = 2\nabla \hat{W}\left(\boldsymbol{C}_{n+\beta_0}\right) \quad \text{for} \quad \beta_0 \in (0,1)} \tag{B.57}$$

Using equation (B.57) with Algorithm 35 gives only first-order time accuracy unless $\beta_0 = 1/2$. To resolve this limitation, Simo and Tarnow [55] proposed the following algorithmic 2nd Piola-Kirchhoff stress which also satisfy equation (B.56):

$$\boxed{\tilde{\boldsymbol{S}} = \nabla \hat{W}\left(\boldsymbol{C}_{n+\beta_0}\right) + \nabla \hat{W}\left(\boldsymbol{C}_{n+(1-\beta_0)}\right) \quad \text{for} \quad \beta_0 \in (0,1)} \tag{B.58}$$

Using equation (B.58) with Algorithm 35 gives second-order time accuracy for any $\beta_0 = (0,1)$. Hence, with the algorithmic 2nd Piola-Kirchhoff stresses, the exact energy-momentum conserving algorithm within a time step $[t_n, t_{n+1}]$ may be given as follows:

Algorithm 36

Energy-momentum conserving algorithm: Simo-Tarnow framework

$$\begin{aligned} &\int_{\Omega_0} \rho_0 \frac{\boldsymbol{V}_{n+1} - \boldsymbol{V}_n}{\Delta t} \cdot \boldsymbol{\eta} d\Omega_0 + \int_{\Omega_0} \tilde{\boldsymbol{S}} \cdot \boldsymbol{F}^T\left(\boldsymbol{\varphi}_{n+1/2}\right) \cdot \cdot \nabla_0 \boldsymbol{\eta} d\Omega_0 \\ &= \int_{\Omega_0} \rho_0 \boldsymbol{b}_{n+1/2} \cdot \boldsymbol{\eta} d\Omega_0 + \int_{\Gamma_0^\sigma} \bar{\boldsymbol{t}}^0_{n+1/2} \cdot \boldsymbol{\eta} d\Gamma_0^\sigma \end{aligned} \tag{B.59a}$$

$$\int_{\Omega_0} \frac{\boldsymbol{\varphi}_{n+1} - \boldsymbol{\varphi}_n}{\Delta t} d\Omega_0 = \int_{\Omega_0} \boldsymbol{V}_{n+1/2} d\Omega_0 \tag{B.59b}$$

$\forall \boldsymbol{\eta} \in \mathcal{V}$, *where the algorithmic 2nd Piola-Kirchhoff stress* $\tilde{\boldsymbol{S}}$ *is given by*

$$\textit{Option 1: } \tilde{\boldsymbol{S}} = \tilde{\boldsymbol{S}}_1 := 2\nabla \hat{W}\left(\boldsymbol{C}_{n+\beta_0}\right) \tag{B.59c}$$

$$\textit{Option 2: } \tilde{\boldsymbol{S}} = \tilde{\boldsymbol{S}}_2 := \nabla \hat{W}\left(\boldsymbol{C}_{n+\beta_0}\right) + \nabla \hat{W}\left(\boldsymbol{C}_{n+(1-\beta_0)}\right) \tag{B.59d}$$

with $\beta_0 \in (0,1)$ *such that*

$$\hat{W}\left(\boldsymbol{C}_{n+1}\right) - \hat{W}\left(\boldsymbol{C}_n\right) = \tilde{\boldsymbol{S}} : \tfrac{1}{2}\left[\boldsymbol{C}_{n+1} - \boldsymbol{C}_n\right] \tag{B.59e}$$

Remark 37 (Algorithm 36)

1. *Accuracy in time: The algorithmic 2nd Piola-Kirchhoff stress* $\tilde{\boldsymbol{S}}$*: Both Option 1, equation* (B.59c)*, and Option 2, equation* (B.59d)*, satisfy equation* (B.59e)*. Although second-order time accuracy is achieved for any* $\beta_0 \in (0,1)$ *in Option 2, it is given only when* $\beta_0 = 1/2$ *in Option 1; otherwise, only first-order time accuracy for* $\beta_0 \neq 1/2$ *in Option 1.*

2. *For the Saint Venant-Kirchhoff materials, we always have* $\boxed{\beta_0 = 1/2}$.

3. *Algorithm 36 is unconditionally stable in the sense of exact energy conservation within a time step. Remember the total energy defines the Lyapunov function of the dynamical system.*

4. *During the implementaion of the fully discretized form (see equations* (B.63)*) of Algorithm 36 via finite-element space discretization, an iterative method, such as the Newton-Raphson method, needs to be used every time step to find* β_0 *which satisfy equation* (B.59e)*. Dealing with* β_0 *calculated at each quadrature point and* $\boldsymbol{\varphi}_{n+1}$ *consistently when computing the tangent stiffness matrix is not guaranteed in Algorithm 36, which may cause an equilibrium convergence issue. For this issue, Algorithm 36 has been implemented only for the Saint Venant-Kirchhoff materials in the paper [55]. For the solution procedure to resolve the issue, see Laursen & Meng [67].*

5. *During the implementation of the fully discretized form of Algorithm 36 via finite-element space discretization by means of the Newton-Raphson*

method, the tangent stiffness matrix becomes asymmetric in general; however, for a sufficient small Δt, it tends to become close to a symmetric matrix.

By the Galerkin projection of infinite dimensional state space

$$C = \left\{ (\boldsymbol{\varphi}, \boldsymbol{V}) : \Omega_0 \to \mathbb{R}^{\mathrm{n_{dim}}} \times \mathbb{R}^{\mathrm{n_{dim}}} \mid \boldsymbol{\varphi} \in Q \text{ and } \boldsymbol{V} \mid_{\Gamma_0^{\boldsymbol{\varphi}}} = \boldsymbol{0} \right\} \tag{B.60}$$

onto a finite dimensional state subspace $C^h \subset C$ via

$$\boldsymbol{\varphi}^h (\boldsymbol{X}, t) = \sum_{A=1}^{\mathrm{n_{node}}} \boldsymbol{N}_A (\boldsymbol{X}) \, \boldsymbol{\varphi}^A (t) \tag{B.61}$$

$$\boldsymbol{V}^h (\boldsymbol{X}, t) = \sum_{A=1}^{\mathrm{n_{node}}} \boldsymbol{N}_A (\boldsymbol{X}) \, \boldsymbol{V}^A (t) \tag{B.62}$$

where $\boldsymbol{N}_A : \Omega_0 \to \mathbb{R}$ $(A = 1, 2, \ldots, \mathrm{n_{node}})$ denotes the finite element interpolation function over an element with $\mathrm{n_{node}}$ nodes, Algorithm 36 may be descretized in space as follows:

$$\sum_{B=1}^{\mathrm{n_{node}}} \boldsymbol{M}_{AB} \frac{\boldsymbol{V}^B_{n+1} - \boldsymbol{V}^B_n}{\Delta t} + \boldsymbol{f}^{\mathrm{int}}_{A\, n+1/2} = \boldsymbol{f}^{\mathrm{ext}}_{A\, n+1/2} \tag{B.63a}$$

$$\int_{\Omega_0} \frac{\boldsymbol{\varphi}^A_{n+1} - \boldsymbol{\varphi}^A_n}{\Delta t} d\Omega_0 = \int_{\Omega_0} \boldsymbol{V}^A_{n+1/2} d\Omega_0 \tag{B.63b}$$

where

$$\boldsymbol{f}^{\mathrm{int}}_{A\, n+1/2} = \int_{\Omega_0} \boldsymbol{B}^T_A \left[\tilde{\boldsymbol{S}}^h \cdot \boldsymbol{F}^T \left(\boldsymbol{\varphi}^h_{n+1/2} \right) \right] d\Omega_0 \tag{B.63c}$$

$$\boldsymbol{f}^{\mathrm{ext}}_{A\, n+1/2} = \int_{\Omega_0} \boldsymbol{N}_A \rho_0 \boldsymbol{b}_{n+1/2} d\Omega_0 + \int_{\Gamma_0^\sigma} \boldsymbol{N}_A \bar{\boldsymbol{t}}^0_{n+1/2} d\Gamma_0^\sigma \tag{B.63d}$$

for $A = 1, 2, \ldots, \mathrm{n_{node}}$. The mass matrix and the so-called B-matrix are given by $\boldsymbol{M}_{AB} = \int_{\Omega_0} \rho_0 \boldsymbol{N}_A \boldsymbol{N}_B d\Omega_0 = \boldsymbol{M}_{BA}$ and $\boldsymbol{B}_A = \nabla_0 \boldsymbol{N}_A$, respectively. Note the finite element subspace of admissible test functions may be defined as

$$\mathcal{V}^h = \left\{ \boldsymbol{\eta}^h \in \mathcal{V} \mid \boldsymbol{\eta}^h = \sum_{A=1}^{\mathrm{n_{node}}} \boldsymbol{N}_A (\boldsymbol{X}) \, \boldsymbol{c}^A, \ \boldsymbol{c}^A \in \mathbb{R}^{\mathrm{n_{node}}} \text{ for } A = 1, 2, \ldots, \mathrm{n_{node}} \right\} \subset \mathcal{V} \tag{B.64}$$

Even after the Galerkin finite element space discretization, all the algorithmic properties such as discrete linear and angular momenta conservations and energy conservation are still preserved under the same conditions for elastodynamics. Consequently, since the projected algorithmic 2nd Piola-Kirchhoff stress $\tilde{\boldsymbol{S}}^h$ must be symmetric, but can be arbitrary for the general smooth elastodynamics, we use the following expressions:

$$\tilde{\boldsymbol{S}}^h := 2\nabla\hat{W}\left(\boldsymbol{C}^h_{n+\beta_0}\right) \quad \text{for} \quad \beta_0 \in (0,1) \tag{B.65}$$

for first-order time accuracy and

$$\tilde{\boldsymbol{S}}^h := \nabla\hat{W}\left(\boldsymbol{C}^h_{n+\beta_0}\right) + \nabla\hat{W}\left(\boldsymbol{C}^h_{n+(1-\beta_0)}\right) \quad \text{for} \quad \beta_0 \in (0,1) \tag{B.66}$$

for second-order time accuracy, respectively in the general hyperelastic elastodynamic model. The numerical implementation aspect of the enegy-momentum conserving algorithm (Simo-Tarnow Framework) with equation (B.66) may be summarized as shown below. Note the Newton-Raphson method has been employed to iteratively solve for the nonlinear elastodynamic equations.

`Numerical Implementation Aspect: Simo-Tarnow Framework`

`Step 1:`

Given the initial conditions $(\boldsymbol{\varphi}^A_n, \boldsymbol{V}^A_n)$ at nodal points for $A = 1, 2, \ldots, \mathrm{n_{node}}$. At the beginning of $n+1$ time level, predict ${}^k\boldsymbol{\varphi}^A_{n+1} = \boldsymbol{\varphi}^A_n$ and ${}^k\boldsymbol{V}^A_{n+1} = \boldsymbol{V}^A_n$, where $k = 1, 2, \ldots, \mathrm{n_{iter}}$ denotes the number of iteration.

`Step 2:`

Compute the right Cauchy-Green strain tensors:

$$\boldsymbol{C}^h_n := \boldsymbol{F}^T(\boldsymbol{\varphi}^h_n)\cdot\boldsymbol{F}(\boldsymbol{\varphi}^h_n) \;\text{ and }\; {}^k\boldsymbol{C}^h_{n+1} := \boldsymbol{F}^T({}^k\boldsymbol{\varphi}^h_{n+1})\cdot\boldsymbol{F}({}^k\boldsymbol{\varphi}^h_{n+1}) \tag{B.67}$$

from the projected deformation gradient given by

$$\boldsymbol{F}(\boldsymbol{\varphi}^h_{n+\alpha}) = \sum_{A=1}^{\mathrm{n_{node}}} \boldsymbol{\varphi}^A_{n+\alpha} \otimes \nabla_0 \boldsymbol{N}_A \tag{B.68}$$

`Step 3:`

Solve for $\beta_0 \in (0,1)$ as close to 0.5 as possible from the residual $r : (0,1) \to \mathbb{R} : \beta \mapsto r(\beta)$ by projected form of equation (B.59e) with equation (B.66), i.e.,

$$r(\beta) := \hat{W}({}^k\boldsymbol{C}^h_{n+1}) - \hat{W}(\boldsymbol{C}_n) - \tfrac{1}{2}\tilde{\boldsymbol{S}}^h : [{}^k\boldsymbol{C}^h_{n+1} - \boldsymbol{C}_n] \tag{B.69}$$

where

$$\tilde{\boldsymbol{S}}^h := \nabla\hat{W}(\boldsymbol{C}^h_{n+\beta}) + \nabla\hat{W}(\boldsymbol{C}^h_{n+(1-\beta)}) \tag{B.70}$$

Notice β_0 is calculated for each quadrature point.

`Step 4:`

Using $\beta = \beta_0$, evaluate the algorithmic 2nd Piola-Kirchhoff stress ${}^k\tilde{\boldsymbol{S}}^h$ and the convected moduli from

$$\begin{aligned} {}^k\tilde{\boldsymbol{S}}^h &:= \nabla\hat{W}({}^k\boldsymbol{C}^h_{n+\beta_0}) + \nabla\hat{W}({}^k\boldsymbol{C}^h_{n+(1-\beta_0)}) && \text{(B.71)} \\ {}^k\mathbb{C}^h &:= 2\beta_0\nabla^2\hat{W}({}^k\boldsymbol{C}^h_{n+\beta_0}) + 2(1-\beta_0)\nabla^2\hat{W}({}^k\boldsymbol{C}^h_{n+(1-\beta_0)}) && \text{(B.72)} \end{aligned}$$

respectively.

`Step 5:`

Solve for ${}^{k+1}\Delta\boldsymbol{\varphi}^B_{n+1}$ by means of the Newton-Raphson method:

$${}^k\boldsymbol{R}_A = -\sum_{B=1}^{\mathrm{n_{node}}} \left[\tfrac{2}{\Delta t^2}\boldsymbol{M}_{AB} + {}^k\boldsymbol{K}_{AB}\right] {}^{k+1}\Delta\boldsymbol{\varphi}^B_{n+1} \tag{B.73}$$

where the residual is given by

$${}^k\boldsymbol{R}_A := \frac{2}{\Delta t^2}\sum_{B=1}^{\mathrm{n_{node}}} \boldsymbol{M}_{AB}\left[{}^k\boldsymbol{\varphi}^B_{n+1} - \boldsymbol{\varphi}^B_n - \Delta t\boldsymbol{V}^B_n\right] + {}^k\boldsymbol{f}^{\mathrm{int}}_{A\,n+1/2} - {}^k\boldsymbol{f}^{\mathrm{ext}}_{A\,n+1/2} \tag{B.74}$$

where ${}^k\boldsymbol{f}^{\mathrm{int}}_{A\,n+1/2}$ and ${}^k\boldsymbol{f}^{\mathrm{ext}}_{A\,n+1/2}$ may be given by equations (B.63c) and (B.63d), respectively. The *asymmetric* (because of the material tangent stiffness) tangent stiffness matrix, i.e., the exact linearization of the internal force, ${}^k\boldsymbol{K}_{AB}$ is given by

$${}^k\boldsymbol{K}_{AB} = {}^k\boldsymbol{K}^{\mathrm{Geo.}}_{AB} + {}^k\boldsymbol{K}^{\mathrm{Mat.}}_{AB} \tag{B.75}$$

where the geometrical and material tangent stiffness matrices are given by

$$
{}^{k}\boldsymbol{K}_{AB}^{\text{Geo.}} = \int_{\Omega_0} \nabla_0 \boldsymbol{N}_A \cdot {}^{k}\tilde{\boldsymbol{S}}^{h} \nabla_0 \boldsymbol{N}_B d\Omega_0 \tag{B.76}
$$

$$
{}^{k}\boldsymbol{K}_{AB}^{\text{Mat.}} = \int_{\Omega_0} \left(\boldsymbol{F}^T({}^{k}\boldsymbol{\varphi}_{n+1/2}^{h}) \cdot \nabla_0 \boldsymbol{N}_A \right) {}^{k}\mathbb{C}^{h} \left(\boldsymbol{F}^T({}^{k}\boldsymbol{\varphi}_{n+\beta_0}^{h}) \cdot \nabla_0 \boldsymbol{N}_B \right) d\Omega_0 \tag{B.77}
$$

respectively. Then correct the solution as

$$
{}^{k+1}\boldsymbol{\varphi}_{n+1}^{A} = {}^{k}\boldsymbol{\varphi}_{n+1}^{A} + {}^{k+1}\Delta\boldsymbol{\varphi}_{n+1}^{A} \tag{B.78}
$$

until the solution converges: $\| \, {}^{k}\boldsymbol{R}_A \, \| \leq$ tolerance.
Return to `Step 1`.

B.2.2 Laursen-Meng Implementation

In order to resolve the convergence issue stated in Remark 37-4, Laursen & Meng [67] proposed a new solution procedure to improve the stress update of Algorithm 36; see [67]. From Algorithm 36, construct the following coupled system of equations:

$$
\begin{aligned}
G\left(\beta, \boldsymbol{\varphi}_{n+1}, \boldsymbol{\eta}\right) &:= \int_{\Omega_0} \rho_0 \frac{\boldsymbol{V}_{n+1} - \boldsymbol{V}_n}{\Delta t} \cdot \boldsymbol{\eta} d\Omega_0 + \int_{\Omega_0} \tilde{\boldsymbol{S}} \cdot \boldsymbol{F}^T\left(\boldsymbol{\varphi}_{n+1/2}\right) \cdot \cdot \nabla_0 \boldsymbol{\eta} d\Omega_0 \\
&\quad - \int_{\Omega_0} \rho_0 \boldsymbol{b}_{n+1/2} \cdot \boldsymbol{\eta} d\Omega_0 - \int_{\Gamma_0^{\sigma}} \bar{\boldsymbol{t}}_{n+1/2}^{0} \cdot \boldsymbol{\eta} d\Gamma_0^{\sigma} \qquad \text{(B.79)} \\
g\left(\beta, \boldsymbol{\varphi}_{n+1}\right) &:= \tfrac{1}{2}\tilde{\boldsymbol{S}} : \left[\boldsymbol{C}_{n+1} - \boldsymbol{C}_n\right] - \hat{W}\left(\boldsymbol{C}_{n+1}\right) + \hat{W}\left(\boldsymbol{C}_n\right) \qquad \text{(B.80)}
\end{aligned}
$$

with the updates:

$$
\boldsymbol{\varphi}_{n+1/2} := \frac{\boldsymbol{\varphi}_{n+1} + \boldsymbol{\varphi}_n}{2} \in Q \tag{B.81}
$$

$$
\boldsymbol{V}_{n+1/2} := \frac{\boldsymbol{V}_{n+1} + \boldsymbol{V}_n}{2} = \frac{\boldsymbol{\varphi}_{n+1} - \boldsymbol{\varphi}_n}{\Delta t} \tag{B.82}
$$

In the numerical implementation employing the Newton-Raphson method, solve the following linearized constraint equation at each point

$$
g\left(\beta, \boldsymbol{\varphi}_{n+1}\right) + \mathrm{D}g\left(\beta, \boldsymbol{\varphi}_{n+1}\right)[\Delta\beta] + \mathrm{D}g\left(\beta, \boldsymbol{\varphi}_{n+1}\right)[\Delta\boldsymbol{\varphi}] = 0 \tag{B.83}
$$

where the directional derivative of g in the direction of $\Delta\beta$ and $\Delta\boldsymbol{\varphi}$ are given as

$$\mathrm{D}g[\Delta\beta] := \left.\frac{d}{d\epsilon}\right|_{\epsilon=0} g\left(\beta+\epsilon\Delta\beta,\boldsymbol{\varphi}\right) \tag{B.84}$$

$$\mathrm{D}g[\Delta\boldsymbol{\varphi}] := \left.\frac{d}{d\epsilon}\right|_{\epsilon=0} g\left(\beta,\boldsymbol{\varphi}+\epsilon\Delta\boldsymbol{\varphi}\right) \tag{B.85}$$

respectively. For the first-order time accurate algorithm, $\mathrm{D}g[\Delta\beta]$ and $\mathrm{D}g[\Delta\boldsymbol{\varphi}]$ are given by

$$\mathrm{D}g[\Delta\beta] = \Delta\boldsymbol{C} : \nabla^2\hat{W}(\boldsymbol{C}_{n+\beta})\Delta\boldsymbol{C}\cdot\Delta\beta \tag{B.86a}$$

$$\begin{aligned}\mathrm{D}g[\Delta\boldsymbol{\varphi}] &= \tfrac{1}{2}\Delta\boldsymbol{C} : \mathbb{C}[\boldsymbol{F}^T\left(\boldsymbol{\varphi}_{n+1}\right)\nabla_0\Delta\boldsymbol{\varphi}] \\ &\quad +[\tilde{\boldsymbol{S}}-\boldsymbol{S}_{n+1}] : [\boldsymbol{F}^T\left(\boldsymbol{\varphi}_{n+1}\right)\nabla_0\Delta\boldsymbol{\varphi}]\end{aligned} \tag{B.86b}$$

respectively, where $\Delta\boldsymbol{C} := \boldsymbol{C}_{n+1}-\boldsymbol{C}_n$ and the convected moduli are given by

$$\mathbb{C} = 4\beta\nabla^2\hat{W}(\boldsymbol{C}_{n+\beta}) \tag{B.86c}$$

For the second-order time accurate algorithm, $\mathrm{D}g[\Delta\beta]$ and $\mathrm{D}g[\Delta\boldsymbol{\varphi}]$ are given by

$$\mathrm{D}g[\Delta\beta] = \tfrac{1}{2}\Delta\boldsymbol{C} : \left[\nabla^2\hat{W}(\boldsymbol{C}_{n+\beta})-\nabla^2\hat{W}(\boldsymbol{C}_{n+(1-\beta)})\right]\Delta\boldsymbol{C}\cdot\Delta\beta \tag{B.87a}$$

$$\begin{aligned}\mathrm{D}g[\Delta\boldsymbol{\varphi}] &= \tfrac{1}{2}\Delta\boldsymbol{C} : \mathbb{C}[\boldsymbol{F}^T\left(\boldsymbol{\varphi}_{n+1}\right)\nabla_0\Delta\boldsymbol{\varphi}] \\ &\quad +[\tilde{\boldsymbol{S}}-\boldsymbol{S}_{n+1}] : [\boldsymbol{F}^T\left(\boldsymbol{\varphi}_{n+1}\right)\nabla_0\Delta\boldsymbol{\varphi}]\end{aligned} \tag{B.87b}$$

respectively, where the convected moduli are given by

$$\mathbb{C} = 2\beta\nabla^2\hat{W}(\boldsymbol{C}_{n+\beta})+2(1-\beta)\nabla^2\hat{W}(\boldsymbol{C}_{n+(1-\beta)}) \tag{B.87c}$$

Simultaneously, solve the linearized equation from equation (B.79):

$$G\left(\beta,\boldsymbol{\varphi}_{n+1},\boldsymbol{\eta}\right)+\mathrm{D}G\left(\beta,\boldsymbol{\varphi}_{n+1},\boldsymbol{\eta}\right)[\Delta\beta]+\mathrm{D}G\left(\beta,\boldsymbol{\varphi}_{n+1},\boldsymbol{\eta}\right)[\Delta\boldsymbol{\varphi}] = 0 \tag{B.88}$$

where the directional derivative of G in the direction of $\Delta\beta$ and $\Delta\boldsymbol{\varphi}$ are given as

$$\mathrm{D}G[\Delta\beta] := \left.\frac{d}{d\epsilon}\right|_{\epsilon=0} G\left(\beta+\epsilon\Delta\beta,\boldsymbol{\varphi}\right) \tag{B.89}$$

$$\mathrm{D}G[\Delta\boldsymbol{\varphi}] := \left.\frac{d}{d\epsilon}\right|_{\epsilon=0} G\left(\beta,\boldsymbol{\varphi}+\epsilon\Delta\boldsymbol{\varphi}\right) \tag{B.90}$$

respectively. For the first-order algorithm,

$$\mathrm{D}G\left(\beta,\boldsymbol{\varphi}_{n+1},\boldsymbol{\eta}\right)[\Delta\beta]=\int_{\Omega_0}2\nabla_0\boldsymbol{\eta}:\left[\boldsymbol{F}(\boldsymbol{\varphi}_{n+1/2})\nabla^2\hat{W}(\boldsymbol{C}_{n+\beta})\Delta\boldsymbol{C}\right]\cdot\Delta\beta d\Omega_0 \tag{B.91a}$$

$$\begin{aligned}\mathrm{D}G\left(\beta,\boldsymbol{\varphi}_{n+1},\boldsymbol{\eta}\right)[\Delta\boldsymbol{\varphi}]=&\int_{\Omega_0}\frac{2\rho_0}{\Delta t^2}\boldsymbol{\eta}\cdot\Delta\boldsymbol{\varphi}d\Omega_0+\int_{\Omega_0}\frac{1}{2}\nabla_0\boldsymbol{\eta}:\left[\nabla_0\Delta\boldsymbol{\varphi}\ \tilde{\boldsymbol{S}}\right]d\Omega_0\\&+\int_{\Omega_0}\nabla_0\boldsymbol{\eta}:\left[\boldsymbol{F}(\boldsymbol{\varphi}_{n+1/2})\mathbb{C}\boldsymbol{F}^T\left(\boldsymbol{\varphi}_{n+1}\right)\nabla_0\Delta\boldsymbol{\varphi}\right]d\Omega_0\end{aligned} \tag{B.91b}$$

where the convected moduli are given by

$$\mathbb{C}=4\beta\nabla^2\hat{W}(\boldsymbol{C}_{n+\beta}) \tag{B.91c}$$

For the second-order algorithm,

$$\begin{aligned}&\mathrm{D}G\left(\beta,\boldsymbol{\varphi}_{n+1},\boldsymbol{\eta}\right)[\Delta\beta]\\&=\int_{\Omega_0}\nabla_0\boldsymbol{\eta}:\left[\boldsymbol{F}(\boldsymbol{\varphi}_{n+1/2})\left(\nabla^2\hat{W}(\boldsymbol{C}_{n+\beta})-\nabla^2\hat{W}(\boldsymbol{C}_{n+(1-\beta)})\right)\Delta\boldsymbol{C}\right]\cdot\Delta\beta d\Omega_0\end{aligned} \tag{B.92a}$$

$$\begin{aligned}&\mathrm{D}G\left(\beta,\boldsymbol{\varphi}_{n+1},\boldsymbol{\eta}\right)[\Delta\boldsymbol{\varphi}]\\&=\int_{\Omega_0}\frac{2\rho_0}{\Delta t^2}\boldsymbol{\eta}\cdot\Delta\boldsymbol{\varphi}d\Omega_0+\int_{\Omega_0}\frac{1}{2}\nabla_0\boldsymbol{\eta}:\left[\nabla_0\Delta\boldsymbol{\varphi}\ \tilde{\boldsymbol{S}}\right]d\Omega_0\\&+\int_{\Omega_0}\nabla_0\boldsymbol{\eta}:\left[\boldsymbol{F}(\boldsymbol{\varphi}_{n+1/2})\mathbb{C}\boldsymbol{F}^T\left(\boldsymbol{\varphi}_{n+1}\right)\nabla_0\Delta\boldsymbol{\varphi}\right]d\Omega_0\end{aligned} \tag{B.92b}$$

where the convected moduli are given by

$$\mathbb{C}=2\beta\nabla^2\hat{W}(\boldsymbol{C}_{n+\beta})+2(1-\beta)\nabla^2\hat{W}(\boldsymbol{C}_{n+(1-\beta)}) \tag{B.92c}$$

See [67] for the details of the so-called "element level β" and "Quadrature Point $\boldsymbol{\beta}$" implementations.

B.2.3 Gonzalez Framework

A significant improvement of the Simo-Tarnow general framework [55] of the exact energy-momentum conserving algorithm for (compressible) elastodynamics has

been proposed by Gonzalez [61] by applying so-called *discrete derivative* [69] to the evaluation of the algorithmic stress. In this framework, an algorithmic stress field $\tilde{\boldsymbol{S}}\left(\boldsymbol{\varphi}_n, \boldsymbol{\varphi}_{n+1}\right)$ is defined as

$$\tilde{\boldsymbol{S}}\left(\boldsymbol{\varphi}_n, \boldsymbol{\varphi}_{n+1}\right) := 2\mathrm{d}\hat{W}\left(\boldsymbol{C}_n, \boldsymbol{C}_{n+1}\right) \tag{B.93}$$

where $\hat{W}\left(\boldsymbol{C}\right) \equiv \bar{W}\left(\boldsymbol{E}\right)$ is the stored energy potential, and "d" denotes a discrete derivative. With the proposition, equation (B.93) may be written as

$$\tilde{\boldsymbol{S}}\left(\boldsymbol{\varphi}_n, \boldsymbol{\varphi}_{n+1}\right) = 2D\hat{W}\left(\boldsymbol{C}_{n+1/2}\right) + 2\frac{\hat{W}\left(\boldsymbol{C}_{n+1}\right) - \hat{W}\left(\boldsymbol{C}_n\right) - D\hat{W}\left(\boldsymbol{C}_{n+1/2}\right) : \Delta\boldsymbol{C}}{\parallel \Delta\boldsymbol{C} \parallel^2}\Delta\boldsymbol{C} \tag{B.94}$$

where $\Delta\boldsymbol{C} := \boldsymbol{C}_{n+1} - \boldsymbol{C}_n$ and $\parallel \Delta\boldsymbol{C} \parallel := \sqrt{\Delta\boldsymbol{C} : \Delta\boldsymbol{C}}$. In terms of the second Piola-Kirchhoff tensor $\boldsymbol{E}$, equation (B.94) can be written as

$$\tilde{\boldsymbol{S}}\left(\boldsymbol{\varphi}_n, \boldsymbol{\varphi}_{n+1}\right) = D\hat{W}\left(\boldsymbol{E}_{n+1/2}\right) + \frac{\hat{W}\left(\boldsymbol{E}_{n+1}\right) - \hat{W}\left(\boldsymbol{E}_n\right) - 2D\hat{W}\left(\boldsymbol{E}_{n+1/2}\right) : \Delta\boldsymbol{E}}{2 \parallel \Delta\boldsymbol{E} \parallel^2}\Delta\boldsymbol{E} \tag{B.95}$$

where $\Delta\boldsymbol{E} := \boldsymbol{E}_{n+1} - \boldsymbol{E}_n$ and $\parallel \Delta\boldsymbol{E} \parallel := \sqrt{\Delta\boldsymbol{E} : \Delta\boldsymbol{E}}$. For the Saint Venant-Kirchhoff materials, the stored energy potential is given by

$$\bar{W} = \int \boldsymbol{S} : d\boldsymbol{E} = \int \mathbb{C} : \boldsymbol{E} : d\boldsymbol{E} = \frac{1}{2}\boldsymbol{E} : \mathbb{C} : \boldsymbol{E} \tag{B.96}$$

where $\mathbb{C}$ is the constant fourth-order tensor of elastic moduli which is major symmetric and positive-definite. Therefore, the algorithmic stress field for this special material model can be reduced as follows:

$$\tilde{\boldsymbol{S}}\left(\boldsymbol{\varphi}_n, \boldsymbol{\varphi}_{n+1}\right) = \mathbb{C} : \boldsymbol{E}_{n+1/2} \tag{B.97}$$

Hence, using the algorithmic 2nd Piola-Kirchhoff stress $\tilde{\boldsymbol{S}}$, the conserving scheme proposed by Gonzalez may be summarized as follows:

Algorithm 37

Gonzalez framework

$$\int_{\Omega_0} \rho_0 \frac{\boldsymbol{V}_{n+1} - \boldsymbol{V}_n}{\Delta t} \cdot \boldsymbol{\eta} d\Omega_0 + \int_{\Omega_0} \tilde{\boldsymbol{S}} \cdot \boldsymbol{F}^T \left(\boldsymbol{\varphi}_{n+1/2}\right) \cdot \cdot \nabla_0 \boldsymbol{\eta} d\Omega_0$$
$$= \int_{\Omega_0} \rho_0 \boldsymbol{b}_{n+1/2} \cdot \boldsymbol{\eta} d\Omega_0 + \int_{\Gamma_0^\sigma} \bar{\boldsymbol{t}}^0_{n+1/2} \cdot \boldsymbol{\eta} d\Gamma_0^\sigma \tag{B.98a}$$
$$\int_{\Omega_0} \frac{\boldsymbol{\varphi}_{n+1} - \boldsymbol{\varphi}_n}{\Delta t} d\Omega_0 = \int_{\Omega_0} \boldsymbol{V}_{n+1/2} d\Omega_0 \tag{B.98b}$$

$\forall \boldsymbol{\eta} \in \mathcal{V}$, *where the algorithmic 2nd Piola-Kirchhoff stress* $\tilde{\boldsymbol{S}}$ *is given by*

$$\tilde{\boldsymbol{S}} \left(\boldsymbol{\varphi}_n, \boldsymbol{\varphi}_{n+1}\right) := 2\mathrm{d}\hat{W} \left(\boldsymbol{C}_n, \boldsymbol{C}_{n+1}\right)$$
$$= 2D\hat{W} \left(\boldsymbol{C}_{n+1/2}\right) + 2\frac{\hat{W} \left(\boldsymbol{C}_{n+1}\right) - \hat{W} \left(\boldsymbol{C}_n\right) - D\hat{W} \left(\boldsymbol{C}_{n+1/2}\right) : \Delta \boldsymbol{C}}{\parallel \Delta \boldsymbol{C} \parallel^2} \Delta \boldsymbol{C} \tag{B.98c}$$

where $\Delta \boldsymbol{C} := \boldsymbol{C}_{n+1} - \boldsymbol{C}_n$ *and* $\parallel \Delta \boldsymbol{C} \parallel := \sqrt{\Delta \boldsymbol{C} : \Delta \boldsymbol{C}}$.

Remark 38 (Algorithm 37)

1. *The algorithmic 2nd Piola-Kirchhoff stress* $\tilde{\boldsymbol{S}}$ *satisfies the directionality condition:*
$$\hat{W} \left(\boldsymbol{C}_{n+1}\right) - \hat{W} \left(\boldsymbol{C}_n\right) = \tilde{\boldsymbol{S}} : \tfrac{1}{2} \left[\boldsymbol{C}_{n+1} - \boldsymbol{C}_n\right] \tag{B.99}$$

2. *The algorithmic 2nd Piola-Kirchhoff stress* $\tilde{\boldsymbol{S}}$ *satisfies the consistency condition:*
$$\tilde{\boldsymbol{S}} = 2D\hat{W} \left(\boldsymbol{C}_{n+1/2}\right) + \mathcal{O} \left(\parallel \Delta \boldsymbol{C} \parallel\right) \tag{B.100}$$
as $\parallel \Delta \boldsymbol{C} \parallel \rightarrow 0$.

3. *Since the algorithmic 2nd Piola-Kirchhoff stress* $\tilde{\boldsymbol{S}}$ *given by equation* (B.98c) *is symmetric, the linear and angular momenta are conserved within a time step* $[t_n, t_{n+1}]$.

4. *When* $\Delta \boldsymbol{C} = \boldsymbol{0}$, *we have* $\tilde{\boldsymbol{S}} = 2D\hat{W} \left(\boldsymbol{C}_{n+1/2}\right)$. *Note that there is no singularity issue due to equation* (B.98c).

5. *The extra itteration needed for computing parameter β_0 in the implementation of Algorithm 36 is not required for Algorithm 37.*

`Numerical Implementation Aspect: Gonzalez Framework`

`Step 1:`

Given the initial conditions $(\boldsymbol{\varphi}_n^A, \boldsymbol{V}_n^A)$ at nodal points for $A = 1, 2, \ldots, \mathrm{n_{node}}$. At the beginning of $n+1$ time level, predict ${}^k\boldsymbol{\varphi}_{n+1}^A = \boldsymbol{\varphi}_n^A$ and ${}^k\boldsymbol{V}_{n+1}^A = \boldsymbol{V}_n^A$, where $k = 1, 2, \ldots, \mathrm{n_{iter}}$ denotes the number of itteration.

`Step 2:`

Compute the right Cauchy-Green strain tensors:

$$\boldsymbol{C}_n^h := \boldsymbol{F}^T(\boldsymbol{\varphi}_n^h) \cdot \boldsymbol{F}(\boldsymbol{\varphi}_n^h) \;\text{ and }\; {}^k\boldsymbol{C}_{n+1}^h := \boldsymbol{F}^T({}^k\boldsymbol{\varphi}_{n+1}^h) \cdot \boldsymbol{F}({}^k\boldsymbol{\varphi}_{n+1}^h) \tag{B.101}$$

from the projected deformation gradient given by

$$\boldsymbol{F}(\boldsymbol{\varphi}_{n+\alpha}^h) = \sum_{A=1}^{\mathrm{n_{node}}} \boldsymbol{\varphi}_{n+\alpha}^A \otimes \nabla_0 \boldsymbol{N}_A \tag{B.102}$$

`Step 3:`

Compute the algorithmic 2nd Piola-Kirchhoff stress ${}^k\tilde{\boldsymbol{S}}^h$:

$${}^k\tilde{\boldsymbol{S}}^h = 2D\hat{W}\left({}^k\boldsymbol{C}_{n+1/2}^h\right) + 2\frac{\hat{W}\left({}^k\boldsymbol{C}_{n+1}^h\right) - \hat{W}\left(\boldsymbol{C}_n^h\right) - D\hat{W}\left({}^k\boldsymbol{C}_{n+1/2}^h\right) : {}^k\Delta\boldsymbol{C}^h}{\| \, {}^k\Delta\boldsymbol{C}^h \, \|^2} {}^k\Delta\boldsymbol{C}^h \tag{B.103}$$

where ${}^k\Delta\boldsymbol{C}^h := {}^k\boldsymbol{C}_{n+1}^h - \boldsymbol{C}_n^h$ and ${}^k\boldsymbol{C}_{n+1/2}^h := ({}^k\boldsymbol{C}_{n+1}^h + \boldsymbol{C}_n^h)/2$.

Stability

```
clear all;
close all;
clc;

format long

global OMEGA XI

RHO1 = 1;
RHO2 = 1;
RHO3 = 0;
ialgo = 100;   % options: 100 (U0), 200 (V0)
irep = 5;      % options: 1(d),2(pd),3(v),4(pv),5(a)

OMEGA = 1;
XI = 0.5;
h_min = 1e-3;
% h_min = 1;

[W,opl,dupl] = gssss (ialgo,RHO1,RHO2,RHO3);

for i=1:80
    h(i) = h_min*(10^((i-1)/10));
end
size_h = size(h)*[0,1]';

for i=1:size_h
    A = Amplification(W,opl,dupl,h(i));
    Eval(:,i) = eig(A);
    RHO(:,i) = abs(Eval(:,i));
    nRHO(:,i) = sort(RHO(:,i));
    OME(i) = h(i)*OMEGA;

    LOGOME(i) = log10(OME(i));
```

```
end

s=0;
for j = 1:size_h/10:size_h
    s = s+1;
    OME1(s) = OME(j);
    nRHO1(:,s) = nRHO(:,j);
end


semilogx(OME1,nRHO1(3,:),'bo',OME1,nRHO1(2,:),'r<',OME1,nRHO1(1,:),'k>',...
    OME,nRHO(3,:),'b-',OME,nRHO(2,:),'r:',OME,nRHO(1,:),'k--')
axis([OME(1) OME(size_h)  0 1.1])
xlabel ('\Omega')
ylabel ('\rho')
grid on;

function [A]=Amplification(W,opl,dupl,h)

global OMEGA XI

LOMEGA = OMEGA*h;
D = opl(6)*W(1)+2*opl(5)*W(2)*XI*LOMEGA+opl(3)*W(3)*LOMEGA*LOMEGA;

alpha = -LOMEGA*LOMEGA/D;
beta = -(2*XI*LOMEGA+opl(1)*W(1)*LOMEGA*LOMEGA)/D;
gamma = 1-(1+2*XI*LOMEGA*opl(4)*W(1)+opl(2)*W(2)*LOMEGA*LOMEGA)/D;

% Form the amplification matrix [A]
A(1,1) = 1+dupl(3)*alpha;
A(2,1) = dupl(5)*alpha;
A(3,1) = alpha;
A(1,2) = dupl(1)+dupl(3)*beta;
A(2,2) = 1+dupl(5)*beta;
A(3,2) = beta;
A(1,3) = dupl(2)-dupl(3)*(1-gamma);
A(2,3) = dupl(4)-dupl(5)*(1-gamma);
```

```
A(3,3) = gamma;
```

Numerical Dissipation and Numerical Dispersion

```
clear all;
close all;
clc;

format long

global OMEGA XI

RHO1 = .4;
RHO2 = .4;
RHO3 = 0.1;
ialgo = 100;   % options: 100 (U0), 200 (V0)
% irep = 5;      % options: 1(d),2(pd),3(v),4(pv),5(a)

OMEGA = 1;
XI = 0;
MASS = 1;
tstart = 0;
h_min = 0;
h_max = 1;
size_h = 80;

un = 1;
vn = 1;
fext = 0;
an = (fext - OMEGA*OMEGA*un - 2*XI*OMEGA*vn)/MASS;


[W,opl,dupl] = gssss (ialgo,RHO1,RHO2,RHO3);

for i=1:size_h
    h(i) = h_min + (i-1)*(h_max-h_min)/size_h;
```

```
A = Amplification(W,opl,dupl,h(i));
COE = poly(A);
if COE(1)<0
    COE = COE*(-1);
end
COE = COE/COE(1);
A1 = -COE(2);
A2 = COE(3);
A3 = -COE(4);

Pc = A2-A1^2/3;
Qc = -A3+A1*A2/3-2*A1^3/27;
Delta = Qc^2/4+Pc^3/27;

alpha1 = -Qc/2+sqrt(Delta);
if alpha1>0
    alpha1 = alpha1^(1/3);
else
    alpha1 = -(-alpha1)^(1/3);
end

alpha2 = -Qc/2-sqrt(Delta) ;
if alpha2>0
    alpha2 = alpha2^(1/3);
else
    alpha2 = -(-alpha2)^(1/3);
end

a(i) = -(alpha1+alpha2)/2+A1/3;
b(i) = sqrt(3)*(alpha1-alpha2)/2;
Phi3(i) = alpha1+alpha2+A1/3;
RHO(i) = sqrt(a(i)*a(i)+b(i)*b(i));


period = b(i)/a(i);
```

```
    Bar_Omega_D(i) =  atan(period);
    if Bar_Omega_D(i)<0
        Bar_Omega_D(i) = pi+Bar_Omega_D(i);
    end

    Bar_Xi(i) = -log(RHO(i))/sqrt(Bar_Omega_D(i)*Bar_Omega_D(i)...
                +log(RHO(i))*log(RHO(i)));

    Bar_Omega(i) = Bar_Omega_D(i)/sqrt(1-Bar_Xi(i)*Bar_Xi(i));
    bOMEGA(i) = Bar_Omega(i)/h(i);

    Pnd(i) = OMEGA/bOMEGA(i)-1;

    % Solution structures
    dn = [un,h(i)*vn,h(i)*h(i)*an]';
    for n=1:2
        A = Amplification(W,opl,dupl,h(i));
        dn = A*dn;
        d(n,i) = dn(1);
    end

    c3(i) = (d(2,i)-2*a(i)*d(1,i)+RHO(i)*RHO(i)*un)/...
            (Phi3(i)*Phi3(i)-2*a(i)*Phi3(i)+RHO(i)*RHO(i));
    c1(i) = un-c3(i);
    c2(i) = (d(1,i)-a(i)*c1(i)-Phi3(i)*c3(i))/b(i);

    % x value
    OME(i) = h(i)*OMEGA/2/pi;
end

s=0;
for j = 1:size_h/10:size_h
    s = s+1;
    OME1(s) = OME(j);
    Bar_Xi1(:,s) = Bar_Xi(:,j);
    Pnd1(s) = Pnd(j);
```

```
end

figure;
plot(OME(2:size_h),Bar_Xi(2:size_h),'b',OME1 ,Bar_Xi1 ,'bs')
if abs(min(Bar_Xi(2:size_h))-max(Bar_Xi(2:size_h)))<1e-11
    axis([OME(2) OME(size_h) min(Bar_Xi(2:size_h))-1 max(Bar_Xi(2:size_h))+1])
else
    axis([OME(2) OME(size_h) min(Bar_Xi(2:size_h)) max(Bar_Xi(2:size_h))])
end
xlabel ('\Omega/2\pi')
ylabel ('Numerical Dissipation')
grid on

figure;
plot(OME(2:size_h),Pnd(2:size_h),'b-',OME1 ,Pnd1 ,'bs')
if abs(min(Pnd(2:size_h))-max(Pnd(2:size_h)))<1e-11
    axis([OME(2) OME(size_h) min(Pnd(2:size_h))-1 max(Pnd(2:size_h))+1])
else
    axis([OME(2) OME(size_h) min(Pnd(2:size_h)) max(Pnd(2:size_h))])
end
xlabel ('\Omega/2\pi')
ylabel ('Numerical Dispersion')
grid on
```

Time History and Order of Convergence

```
clear all;
close all;
clc;

format long

global tstart tend TOL
global mission align
global Time1 Energy1 Momenta1 State1
global Time2 Energy2 Momenta2 State2
global Time3 Energy3 Momenta3 State3
global Time4 Energy4 Momenta4 State4
global Time5 Energy5 Momenta5 State5
global Time6 Energy6 Momenta6 State6
global spectral savefig type  gridon Cond
global Header0 Header1 Header2 Header3 Header4 Problem
global eta1 eta2 option

tstart = 0;
TOL = 1e-8; % Tolerance


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
mission = 1; %1: Time evolution, 2: Time accuracy
type = 1; %1: Implicit GSSSS, 2: PCE GSSSS
eta1 = 1;
eta2 = 1; %PCE
align = 1; %Set 1 to align acceleration for mission 2
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
savefig = 0; % 0: Off, 1: On
gridon = 0; % 0: Off, 1: On
Cond = 0; % 0: Conservative Sys, 1: Dissipative Sys, 2: Forced-Dissipative Sys
Problem = 8;
```

```
Header0 = {'DE','HS','SS','SP','NO','KP','LJ','ST'};
Header1 = {'CON','DIS','FD'};
Header2 = {'TE','LM','AM','TED','LMD','AMD','TER','LMR','AMR'};
Header3 = {'POS','VEL','ACC','R'};

if Problem<5
    SDOFInputParameters;
elseif Problem==5
    NOInputParameters;
elseif Problem==6
    KPInputParameters;
elseif Problem==7
    LJ2ImputParameters;
elseif Problem==8
    STInputParameters;
end

switch mission
    case 1 %Time History
        if eta2 == 1
            h = .01;
            tend = 10;
%             h=1; %tend=100
%             tend = 100;
        elseif eta2 == 0
            h = .001;%DE,SS: .001, SP: .01
            tend = 1; %DE,SS: 1, SP: 10
        end

        option = 4;
        irep = 1;
        spectral = 3;

        if spectral == 1 %U0V0/V0U0 Optimal
            [Time1,Energy1,Momenta1,State1] = MDOF_Analysis(1,1,1,100,irep,h,tend );
            [Time2,Energy2,Momenta2,State2] = MDOF_Analysis(.8,1,.8,100,irep,h ,tend);
```

```
[Time3,Energy3,Momenta3,State3] = MDOF_Analysis(.6,1,0.6,100,irep,h,tend );
[Time4,Energy4,Momenta4,State4] = MDOF_Analysis(.4,1,0.4,100,irep,h,tend );
[Time5,Energy5,Momenta5,State5] = MDOF_Analysis(.2,1,0.2,100,irep,h,tend );
[Time6,Energy6,Momenta6,State6] = MDOF_Analysis(0,1,0,100,irep,h,tend );
elseif spectral == 2
[Time1,Energy1,Momenta1,State1] = MDOF_Analysis(1,1,1,100,irep,h,tend );
[Time2,Energy2,Momenta2,State2] = MDOF_Analysis(1,1,.8,100,irep,h,tend );
[Time3,Energy3,Momenta3,State3] = MDOF_Analysis(1,1,0.6,100,irep,h,tend );
[Time4,Energy4,Momenta4,State4] = MDOF_Analysis(1,1,0.4,100,irep,h,tend );
[Time5,Energy5,Momenta5,State5] = MDOF_Analysis(1,1,0.2,100,irep,h,tend );
[Time6,Energy6,Momenta6,State6] = MDOF_Analysis(1,1,0,100,irep,h,tend );
elseif spectral == 3
[Time1,Energy1,Momenta1,State1] = MDOF_Analysis(1,1,1,200,irep,h,tend );
[Time2,Energy2,Momenta2,State2] = MDOF_Analysis(1,1,.8,200,irep,h,tend );
[Time3,Energy3,Momenta3,State3] = MDOF_Analysis(1,1,0.6,200,irep,h,tend );
[Time4,Energy4,Momenta4,State4] = MDOF_Analysis(1,1,0.4,200,irep,h,tend );
[Time5,Energy5,Momenta5,State5] = MDOF_Analysis(1,1,0.2,200,irep,h,tend );
[Time6,Energy6,Momenta6,State6] = MDOF_Analysis(1,1,0,200,irep,h,tend );
elseif spectral == 4
[Time1,Energy1,Momenta1,State1] = MDOF_Analysis(1,1,0,100,irep,h,tend );
[Time2,Energy2,Momenta2,State2] = MDOF_Analysis(.8,1,0,100,irep,h,tend );
[Time3,Energy3,Momenta3,State3] = MDOF_Analysis(.6,1,0,100,irep,h,tend );
[Time4,Energy4,Momenta4,State4] = MDOF_Analysis(.4,1,0,100,irep,h,tend );
[Time5,Energy5,Momenta5,State5] = MDOF_Analysis(.2,1,0,100,irep,h,tend );
[Time6,Energy6,Momenta6,State6] = MDOF_Analysis(0,1,0,100,irep,h,tend);
elseif spectral == 5
[Time1,Energy1,Momenta1,State1] = MDOF_Analysis(1,1,0,200,irep,h,tend );
[Time2,Energy2,Momenta2,State2] = MDOF_Analysis(.8,1,0,200,irep,h,tend );
[Time3,Energy3,Momenta3,State3] = MDOF_Analysis(.6,1,0,200,irep,h,tend );
[Time4,Energy4,Momenta4,State4] = MDOF_Analysis(.4,1,0,200,irep,h,tend );
[Time5,Energy5,Momenta5,State5] = MDOF_Analysis(.2,1,0,200,irep,h,tend );
[Time6,Energy6,Momenta6,State6] = MDOF_Analysis(0,1,0,200,irep,h,tend );
elseif spectral == 6
[Time1,Energy1,Momenta1,State1] = MDOF_Analysis(1,1,1,100,irep,h,tend );
[Time2,Energy2,Momenta2,State2] = MDOF_Analysis(.8,.8,.8,100,irep,h,tend);
[Time3,Energy3,Momenta3,State3] = MDOF_Analysis(.6,.6,.6,100,irep,h,tend );
```

```
            [Time4,Energy4,Momenta4,State4] = MDOF_Analysis(.4,.4,.4,100,irep,h,tend );
            [Time5,Energy5,Momenta5,State5] = MDOF_Analysis(.2,.2,.2,100,irep,h,tend );
            [Time6,Energy6,Momenta6,State6] = MDOF_Analysis(0,0,0,100,irep,h,tend );
            elseif spectral == 7
            [Time1,Energy1,Momenta1,State1] = MDOF_Analysis(1,1,1,200,irep,h,tend );
            [Time2,Energy2,Momenta2,State2] = MDOF_Analysis(.8,.8,.8,200,irep,h,tend );
            [Time3,Energy3,Momenta3,State3] = MDOF_Analysis(.6,.6,.6,200,irep,h,tend );
            [Time4,Energy4,Momenta4,State4] = MDOF_Analysis(.4,.4,.4,200,irep,h,tend );
            [Time5,Energy5,Momenta5,State5] = MDOF_Analysis(.2,.2,.2,200,irep,h,tend );
            [Time6,Energy6,Momenta6,State6] = MDOF_Analysis(0,0,0,200,irep,h,tend );
        end

        MDOF_Output(h)



    case 2 %Time Accuraty
%         h = [0.0000001 0.000001 0.00001 0.0001]; %tend = .001
        h = [0.000001 0.00001 0.0001 0.001]; %tend = .01
%         h = [0.00001 0.0001 0.001 0.01]; %tend = .1
%         h = [0.00001 0.00005 0.0001 0.0005];
%         h = [0.00001 0.00005 0.0001 0.0002];

        option = 3;
        spectral = 6;

        irep = 5;
        tend = .01 ;

        switch spectral
            case 0 % Input
                ialgo = 100;
                RHO1 = .9;
                RHO2 = .9;
                RHO3 = .9;
            case 1 % MPR-EPA
```

```
        ialgo = 100;
        RHO1 = 1;
        RHO2 = 1;
        RHO3 = 1;
    case 2 % Newmark
        ialgo = 100;
        RHO1 = 1;
        RHO2 = 1;
        RHO3 = 0;
    case 3 % MPR-MPA
        ialgo = 200;
        RHO1 = 1;
        RHO2 = 1;
        RHO3 = 0;
    case 4 %U0V0/V0U0 Optimal
        ialgo = 100;
        RHO1 = 0.8;
        RHO2 = 1;
        RHO3 = 0.8;
    case 5 % U0
        ialgo = 100;
        RHO1 = 0.6;
        RHO2 = 0.9;
        RHO3 = 0.2;
    case 6 % V0
        ialgo = 200;
        RHO1 = 0.6;
        RHO2 = 0.9;
        RHO3 = 0.2;
end



[Time1,Energy1,Momenta1,State1] = MDOF_Analysis(RHO1,RHO2,RHO3,ialgo,irep,h(1),tend );
[Time2,Energy2,Momenta2,State2] = MDOF_Analysis(RHO1,RHO2,RHO3,ialgo,irep,h(2),tend );
[Time3,Energy3,Momenta3,State3] = MDOF_Analysis(RHO1,RHO2,RHO3,ialgo,irep,h(3),tend );
```

```
        [Time4,Energy4,Momenta4,State4] = MDOF_Analysis(RHO1,RHO2,RHO3,ialgo,irep,h(4),tend);

        MDOF_Output(h)

end

function [Time,Energy,Momenta,State] = MDOF_Analysis(RHO1,RHO2,RHO3,ialgo,irep,h,tend)

global M C K Fext0 f0 BETA
global tstart  TOL
global q0 v0 ndim ndof
global mission align type   Problem
global W xsic xsip opl  dupl option step

[ConQ0,Fcon0,Fdis0,Fext0] = EFA(q0,v0,tstart) ;

a0 = M\(Fext0 + Fdis0 - Fcon0); % Initial Acceleration

% Set up the array
step = ceil((tend - tstart)/h);
Time = zeros(2,step);
Energy = zeros(3,step);
Momenta = zeros(2,step);
if Problem < 5
    State = zeros(3,step);
elseif (Problem == 5||Problem ==8)
    State = zeros(9,step);
elseif Problem == 6
    State = zeros(12,step);
elseif Problem == 7
    State = zeros(25,step);
end

istep = 1;
Time(1,istep) = tstart;
Time(2,istep) = tstart;
Energy(:,istep) = ConQ0(1:3);
```

```
Momenta(:,istep) = ConQ0(4:size(ConQ0)*[1 0]');
if Problem<5
    State(:,istep) = [q0 v0 a0]';
elseif (Problem == 5 || Problem == 6 ||  Problem == 8)
    State(:,istep) = [q0' v0' a0' norm(q0) norm(v0) norm(a0) ]' ;
elseif Problem == 7
    State(:,istep) = [q0' v0' a0' ...
        norm(q0(1:ndim)) norm(q0(ndim+1:size(q0)*[1 0]'))...
        norm(v0(1:ndim)) norm(v0(ndim+1:size(q0)*[1 0]'))...
        norm(a0(1:ndim)) norm(a0(ndim+1:size(q0)*[1 0]'))...
            norm(q0(1:ndim)-q0(ndim+1:size(q0)*[1 0])) ]' ;
end


% GSSSS algorithmic paraneters
[W,opl,dupl] = gssss (ialgo,RHO1,RHO2,RHO3);
[xsip,xsic] = getXsi(W, opl, dupl, h, irep);
phi = W(1)*(opl(6)-1);

qn = q0;
vn = v0;
an = a0;
tn = tstart;


steps = ceil((tend-tstart)/h)+1;
while (istep<steps)
    t = tstart + h*istep;
    istep = istep + 1;

    th = (1-W(1))*tn+W(1)*t;
    Fexth = sin(BETA*th)*f0;

    % Predictions
    qp = xsip(1,1)*qn + xsip(1,2)*vn + xsip(1,3)*an;
    qh = xsip(2,1)*qn + xsip(2,2)*vn + xsip(2,3)*an;
```

```
vh = xsip(3,1)*qn + xsip(3,2)*vn + xsip(3,3)*an;
ah = xsip(4,1)*qn + xsip(4,2)*vn + xsip(4,3)*an;


converge = -1;
iteration = 0;
while (converge < 0)
    iteration = iteration +1;

    if iteration > 100
     disp('Caution: Newton-Rapson Method Might Fail!')
    break;
    end

    if Problem<5
        [Fconh,Kt] =  SDOF_FK(h,qn,qp,qh,vn,an);
    elseif Problem == 5
        [Fconh,Kt] =  NO_FK(h,qn,qp,qh,vn,an);
    elseif Problem == 6
        [Fconh,Kt] =  KP_FK(h,qn,qp,qh,vn,an);
    elseif Problem == 7
        [Fconh,Kt] = LJ2_InternalForce(h,qn,qp,qh,vn );
    elseif Problem == 8
        Fconh = K*qh;
        Kt = xsic(2)*K;
    end

    Fdish = -C*vh; % Algorithmic Dissipative force
    res = Fexth + Fdish - M*ah - Fconh;  %Residual
    jac = - xsic(4)*M - xsic(3)*C - Kt;  %Jacobian
    del = (-1)*jac\res;

    if (abs(res)<TOL)
        converge = 1;
    else
        % Corrections
```

```
            qp = qp + xsic(1)*del;
            qh = qh + xsic(2)*del;
            vh = vh + xsic(3)*del;
            ah = ah + xsic(4)*del;
        end
    end

    % Updates
    ap = an + (ah-an)/opl(6)/W(1);
    vp = vn + dupl(4)*an*h + dupl(5)*(ap-an)*h;
    qp = qn + dupl(1)*vn*h + dupl(2)*an*h*h + dupl(3)*(ap-an)*h*h;

%     S1=10; S2=0;
%     EOMp =  M*ap+C*vp+S1*qp*(1+S2*qp^2);


    [ConQ,Fcon,Fdis,Fext] = EFA(qp,vp,t);

    if istep==2
        ap_real = (ap-phi*an)/(1-phi);
    else
        ap_real = (1+phi)*ap-phi*an;
    end

    Time(1,istep) = t;
    Time(2,istep) = phi*Time(1,istep-1)+(1-phi)*Time(1,istep);
    if Problem<5
        State(:,istep) = [qp vp ap]';
    elseif  (Problem == 5 || Problem == 6 || Problem == 8)
        State(:,istep) = [qp' vp' ap' norm(qp) norm(vp) norm(ap) ]' ;
    elseif Problem == 7
        State(:,istep) = [qp' vp' ap' ...
                norm(qp(1:3)) norm(qp(4:6))...
                norm(vp(1:3)) norm(vp(4:6))...
                norm(ap(1:3)) norm(ap(4:6))...
                    norm(qp(1:3)-qp(4:6)) ]' ;
```

```
    end
    Energy(:,istep) = ConQ(1:3);
    Momenta(:,istep) = ConQ(4:size(ConQ)*[1 0]');

    if mission==2
        if istep==steps-1
            as=ap;
        end
    end

    qn = qp;
    vn = vp;
    an = ap;
    tn = t;
end

% Acceleration Alighment %
if mission == 2
    if align == 1
        ap = (-phi*as+(1+phi)*ap);
%         State(13:18,size(Time)*[0 1]')=ap;
        if Problem<5
            State(3,size(Time)*[0 1]') = ap;
        elseif (Problem == 5)
            State(5:6,size(Time)*[0 1]') = ap;
        elseif (Problem == 6)
            State(7:9,size(Time)*[0 1]') = ap;
        elseif Problem == 7
            State(13:18,size(Time)*[0 1]') = ap;
        end
    end
end
%%%%%%%%%%%%%%%%%%%%%%%%%

function [ConQ,Fpot,Fdis,Fext] =  EFA(q,v,t)
global M C f0 BETA alpha sigma
global S1 S2 S EA L g k K
```

```
global Problem

Fdis = -C*v; % Initial Dissipative force
Fext =  sin(BETA*t)*f0;


switch Problem
    case 1 % Duffing
        PE = S1*q^2*(2+S2)/4;
        KE = M*v^2/2;
        TE = PE+KE;
        Fpot = S1*q*(1+S2*q^2);
        LM = M*v;
        ConQ = [TE KE PE LM]';
    case 2 % Hardening Spring
        r =  q;%abs(q);
        PE = 2*(S-EA)*sqrt(L^2+r^2) + (EA/L)*r^2;
        KE = M*v^2/2;
        TE = PE+KE;
        Fpot = (2*(S-EA)/(sqrt(L^2+q^2)))*q + (2*EA/L)*q ;
        PE = -PE;
        LM = M*v;
        ConQ = [TE KE  PE LM]';
    case 3 % Softening Spring
        r = q;%abs(q);
        PE = S*log(cosh(r));
        KE = M*v^2/2;
        TE = PE+KE;
        Fpot = S*tanh(q);
        LM = M*v;
        ConQ = [TE KE PE LM]';
    case 4 % Simple Pendulum
        PE = g*L*(1-cos(q));
        KE = M*v^2/2;
        TE = PE+KE;
        Fpot = g*L*sin(q);
```

```
        LM = M*v;
        ConQ = [TE KE PE LM]';

    case 5 %Nonlinear Oscillator
        r = norm(q);
        PE = r^2*((r^2-1)^2);
        KE = (1/2)*dot(v,M*v);
        TE = PE+KE;
        Fpot = (6*r^4-8*r^2+2)*q;
        p = M*v;
        b =  1;
        AM = b*(q(1)*p(2)-q(2)*p(1));
        ConQ = [TE KE PE norm(p) norm(AM)]';

    case 6 % Kepler Problem
        r = norm(q);
        PE =  -k/r;
        KE = (1/2)*dot(v,M*v);
        TE = PE+KE;
        Fpot =  (k/r^3)*q;
        Fpot =  (k/r^3)*q;

        %LINEAR SYS
%         PE =    dot(q,k*q)/2;
%         KE = (1/2)*dot(v,M*v);
%         TE = PE+KE;
%         Fpot = k*q;

        p = M*v;
        AM = cross(q,p);
        ConQ = [TE KE PE norm(p) norm(AM)]';

    case 7
        r = norm(q(1:3)-q(4:6));
        x = sigma/r;
```

```
        PE = alpha*(x^5-x^3); % Potential Energy
        KE = (1/2)*dot(v,M*v); %  Kinetic Energy
        TE = KE + PE; % Total Energy

        tau = (alpha/sigma)*(3*x^4-5*x^6)*(1/r);
        Fpot = tau*[transpose(q(1:3)-q(4:6)) transpose(q(4:6)-q(1:3))]';
        % Initial Conservative force

        LM = tau*([q(1:3)-q(4:6)]+ [q(4:6)-q(1:3)]);
        p = M*v; % Momentum
        AM1 = cross(q(1:3),p(1:3));
        AM2 = cross(q(4:6),p(4:6));
        AM = AM1 + AM2;

        ConQ = [TE KE PE norm(LM) norm(AM)]';

    case 8
        PE = (1/2)*dot(q,K*q);
        KE = (1/2)*dot(v,M*v);
        TE = KE+PE;

        Fpot = K*q;
        p = M*v;
        b =  1;
        AM = b*(q(1)*p(2)-q(2)*p(1));
        ConQ = [TE KE PE norm(p) norm(AM)]';

end
```